

Prima Prova in Itinere di Calcolatori Elettronici – 07.11.2005

Corso di Laurea di Ing. Gestionale e di Ing. delle Telecomunicazioni

A.A. 2005-2006

Cognome		Nome		Matricola	
---------	--	------	--	-----------	--

Preambolo

Si consideri l'ultima cifra C del numero di matricola del candidato, e siano:

- $X_q = [C/3]$ (parte intera di $C/3$) e siano x_3 e x_2 i bit della sua rappresentazione con x_3 il bit più significativo e x_2 il bit meno significativo;
- $X_m = |C|_4$ (C modulo 4) e siano x_1 e x_0 i bit della sua rappresentazione con x_1 il bit più significativo e x_0 il meno significativo.

Compilare il seguente prospetto perché ne sarà richiesto l'utilizzo negli esercizi successivi:

$C = \underline{\quad}$;

$X_q = [C/3] = \underline{\quad}$ rappresentato da $x_3 = \underline{\quad}$, $x_2 = \underline{\quad}$ quindi $\langle X_q \rangle = \langle \underline{\quad} \underline{\quad} \rangle$;

$X_m = |C|_4 = \underline{\quad}$ rappresentato da $x_1 = \underline{\quad}$, $x_0 = \underline{\quad}$ quindi $\langle X_m \rangle = \langle \underline{\quad} \underline{\quad} \rangle$;

pertanto $\langle X_q X_m \rangle = \langle \underline{\quad} \underline{\quad} \underline{\quad} \underline{\quad} \rangle$

Esercizio 1

Si consideri il numero N dato dal vostro numero di matricola moltiplicato per $10^{-(X_q+X_m+1)}$.

- Si scriva la sua rappresentazione secondo lo standard IEEE/ANSI 754 dei numeri *floating point* su 4 byte.
- Si scriva poi il suo valore in esadecimale.

Esercizio 2

Si consideri una rete combinatoria R_1 avente due ingressi u_3 e u_2 ed una uscita z_1 . L'uscita vale 1 quando $u_3 = x_3$ e $u_2 = x_2$.

Si consideri una rete combinatoria R_2 avente due ingressi u_1 e u_0 ed una uscita z_0 . L'uscita vale 1 quando $u_1 \neq x_1$ e $u_0 \neq x_0$.

Si consideri quindi la rete complessiva R avente come ingressi u_3, u_2, u_1, u_0 e l'uscita $Z = z_1 \text{ OR } z_0$

Si scrivano:

- La tabella di verità di R_1 e R_2 , la loro espressioni logiche SP, e le loro sintesi a porte.
- La tabella di verità per la rete R e la sua Mappa di Karnaugh
- Nella Mappa di Karnaugh si indichino:
 - gli implicant primari;
 - quelli essenziali;
 - la sua copertura minima
 - l'espressione logica SP priva di alee statiche del primo ordine.

Esercizio 3

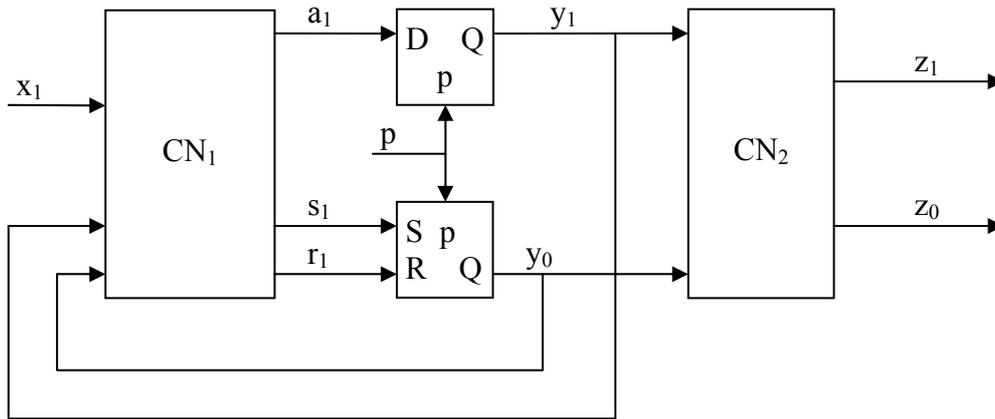
Realizzare un circuito *sequenziale asincrono* che partendo da una situazione di stabilità con $x=0$ e $z=0$ faccia quanto segue: sul piedino $z = \text{MSB}(|\sum \text{valori ricevuti sul piedino } x|_4)$ dove l'operatore MSB restituisce il bit più significativo dell'espressione. Nota: si usi un *cortocircuito* per la sottorete sequenziale asincrona.

Esercizio 4

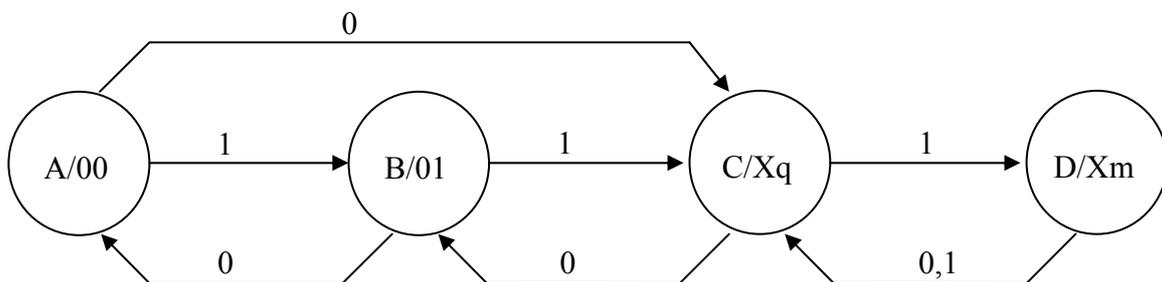
Si disegni il diagramma di flusso di una *rete sequenziale sincrona di Moore* avente due piedini in ingresso i e r e due uscite z_1 e z_0 , avente il seguente comportamento: quando sul piedino r è presente uno 0 entrambe le uscite vengono portate al valore 0. Quando r vale 1 l'uscita z_1 va a 1 quando sul piedino arriva la sequenza di bit identificata da $\langle XqXm \rangle$. Le sequenze sono disgiunte.

Esercizio 5

Si consideri il modello di *rete sequenziale sincrona di Moore* mostrato nella figura seguente:



e si consideri quindi il seguente diagramma di stato:



Sintetizzare la rete descritta, utilizzando il seguente assegnamento per gli stati:

A: $y_1y_0 = 00$,

B: $y_1y_0 = 01$,

C: $y_1y_0 = 11$,

D: $y_1y_0 = 10$.

Seconda Prova in Itinere di Calcolatori Elettronici – 02.12.2005

Corso di Laurea di Ing. Gestionale e di Ing. delle Telecomunicazioni

A.A. 2005-2006

Cognome		Nome		Matricola	
---------	--	------	--	-----------	--

Esercizio 1

Tradurre il seguente listato scritto in linguaggio C in linguaggio Assembler 8086, **disegnando** lo stato dello *stack* dopo ogni chiamata di funzione e **commentando** opportunamente il codice scritto. Nota: i numeri interi sono rappresentati su 2 byte.

```
int n = 9;
int operazione(int j, int k)
{
    int m;
    m = j + k;
    return m;
}
int calcolo(int a, int b)
{
    int s;
    s = operazione(a, b);
    return s;
}
int main()
{
    int i, r;
    int a1 = 1;
    int a2 = 1;
    i = 0;
    while (i < n) {
        r = calcolo(a1, a2);
        i++;
        a2 = a1;
        a1 = r;
    }
    return r;
}
```

Esercizio 2

Si realizzi il programma in linguaggio Assembler 8086 che faccia quanto segue:

1. attenda che l'utente digiti un numero **N** compreso tra **1** e **5** attraverso una tastiera connessa all'interfaccia seriale di ingresso posta agli indirizzi simbolici RBR e RSR (i numeri da '1' a '5' hanno codifiche ASCII da 31H a 35H);
2. si depositi questo numero nella locazione ampia un byte avente nome simbolico **N**;
3. per **N** volte:
 - a. si attenda che l'utente digiti **due** cifre comprese tra **0** e **9** seguite dal tasto 'a';
 - b. interpretando la stringa di caratteri digitati come un numero in base dieci, si memorizzi tale numero nella posizione *i-esima* del vettore collocato a partire dalla locazione **VETT** (**VETT** ha 5 elementi di un byte ciascuno).
4. si memorizzi nella locazione ampia due byte avente nome simbolico **RISU**, la somma degli elementi del vettore **VETT**;
5. si invii sull'interfaccia di uscita i caratteri ASCII necessari a stampare il valore di **RISU**.

Esercizio 3

Si descriva:

- a) l'arbitraggio del bus (aspetti generali);
- b) l'*arbitro asincrono elementare*, indicando il diagramma di flusso e la sintesi per realizzarlo.

Esame scritto di Calcolatori Elettronici – 12.12.2005

Corso di Laurea di Ing. Gestionale e di Ing. delle Telecomunicazioni

A.A. 2005-2006

Cognome		Nome		Matricola	
---------	--	------	--	-----------	--

Istruzioni:

Coloro che intendono recuperare il 1° compitino devono svolgere gli esercizi 1, 2 e 3 (pag. 1).

Coloro che intendono recuperare il 2° compitino devono svolgere gli esercizi 4, 5 e 6 (pag. 2).

Coloro che intendono fare l'intero esame devono svolgere gli esercizi 1, 2, 4, e 5 (pag. 1 e 2).

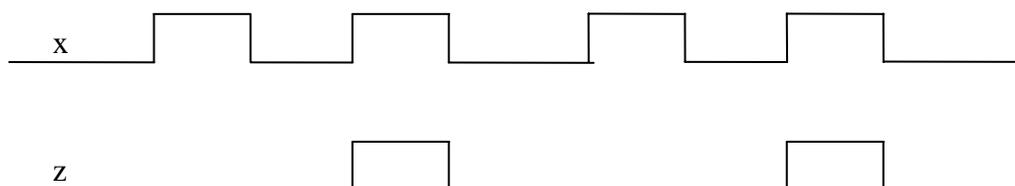
Non è ammesso l'utilizzo materiale didattico o appunti durante questa prova, ad eccezione dell'elenco di istruzioni del processore 8086.

Esercizio 1

Si consideri una Rete Sequenziale Asincrona avente un ingresso x ed una uscita z , che inizialmente è in una situazione di stabilità con $x=0$, e $z=0$. La sua evoluzione temporale è tale che:

1. $z=1$ quando sul piedino x si è presentato per due volte *distinte* il livello logico alto.
2. l'uscita z torna a "0", non appena l'ingresso x torna a livello logico basso.

Esempio:



Per tale rete:

- a) si disegni il corrispondente diagramma di flusso, e
- b) si sintetizzi il suo circuito logico.

Esercizio 2

Si consideri una Rete Sequenziale Sincrona di Moore avente tre ingressi x_2 , x_1 , e x_0 e due uscite z_1 , z_0 . Le due uscite, prese nell'ordine indicato rappresentano, in ciascun istante, ed in binario naturale, il numero Z . Vale la relazione: $Z = |x_2 - x_1 + x_0|_4$, essendo x_2 , x_1 , e x_0 il numero degli "1" logici che si sono presentati, fino all'istante considerato, agli ingressi x_2 , x_1 , e x_0 rispettivamente.

Disegnare il grafo di stato.

Esercizio 3-O

Si descriva il Flip-Flop S-R evidenziandone:

- a) tabella di flusso,
- b) tabelle di applicazione, e
- c) la sintesi a porte NAND.

Esercizio 4

Tradurre il seguente listato scritto in linguaggio C in linguaggio **Assembler 8086**, *disegnando lo stato dello stack dopo ogni chiamata di funzione*, e **commentando** opportunamente il codice scritto. Nota: i numeri interi sono rappresentati su 2 byte.

```
int z;

int uno(int a, int b) {
    int c;

    c = a + b;
    return c;
}

int due(int p, int q) {
    int r;

    r = p - q;
    r = uno(r, p);
    return r;
}

int main() {
    int j, k, x, y;

    j = 1; k = 2; x = 3; y = 4;
    x = uno(j, k);
    y = due(x, j);
    z = x + y;
    return z;
}
```

Esercizio 5

Si realizzi il programma in linguaggio **Assembler 8086** che faccia quanto segue:

1. attenda che l'utente digiti due cifre numeriche attraverso una tastiera connessa all'interfaccia seriale di ingresso posta agli indirizzi simbolici RBR e RSR (i numeri da '0' a '9' hanno codifiche ASCII da 30H a 39H);
2. memorizzi tale numero, interpretato come numero in base 10, in formato binario nella locazione ampia un byte avente nome simbolico **A**;
3. invii sull'interfaccia di uscita i caratteri necessari per *andare a capo* e spostarsi **5** righe più in basso (0x0D è il codice ASCII del ritorno carrello e 0x0A è il codice ASCII dell'avanzamento a capo);
4. quindi invia all'interfaccia di uscita la sequenza:
00 02 04 ... NN-2 NN
 ove **NN** è il numero pari non maggiore del numero **A** (ogni numero è costituito da due caratteri ASCII ed è separato dal successivo da uno spazio).

Esercizio 6-O

Si spieghi a cosa serve il controllore d'interruzione all'interno di un calcolatore e si disegni il suo montaggio all'interno di un calcolatore x86.

Esame scritto di Calcolatori Elettronici – 10.01.2006
Corso di Laurea di Ing. Gestionale e di Ing. delle Telecomunicazioni

A.A. 2005-2006

Cognome		Nome		Matricola	
---------	--	------	--	-----------	--

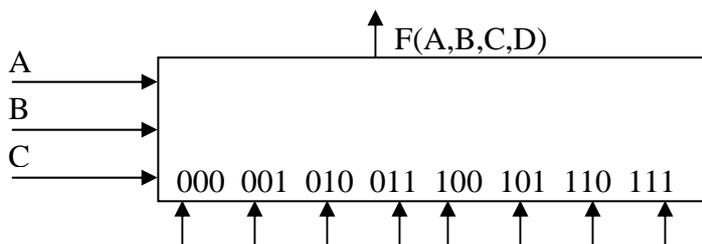
Istruzioni:

Coloro che intendono recuperare il 1° compitino devono svolgere gli esercizi 1, 2 e 3 (pag. 1).
 Coloro che intendono recuperare il 2° compitino devono svolgere gli esercizi 4, 5 e 6 (pag. 2).
 Coloro che intendono fare l'intero esame devono svolgere gli esercizi 1, 2, 4, e 5 (pag. 1 e 2).
 Non è ammesso l'utilizzo materiale didattico o appunti durante questa prova, ad eccezione dell'elenco di istruzioni del processore 8086.

Esercizio 1

Sintetizzare la funzione di 4 variabili non completamente specificate, rappresentata nella tabella, mediante un commutatore 8 ad 1, le cui variabili di selezione siano pilotate dalle variabili A, B, e C rispettivamente (vedi figura).

AB \ CD	00	01	11	10
00	0	1	0	-
01	0	0	-	0
11	1	1	-	0
10	0	0	1	-



Esercizio 2

Realizzare un circuito sequenziale sincronizzato con Flip-Flop J-K che riconosca la sequenza **010** (interallacciata).

Esercizio 3-O

Si descriva il Flip-Flop J-K evidenziandone:

- a) tabella di flusso,
- b) tabelle di applicazione, e
- c) la sua sintesi circuitale.

Esercizio 4

Tradurre il seguente listato scritto in linguaggio C in linguaggio **Assembler 8086**, *disegnando lo stato dello stack dopo ogni chiamata di funzione*, e **commentando** opportunamente il codice scritto. Nota: i numeri interi sono rappresentati su 2 byte.

```
int g1;
int g2;

int due(int p, int q) {
    int r;

    r = p + q;
    return r;
}

int uno(int a, int b) {
    int c;

    c = due(a, b);
    g2 = g2 + c;
    return g2;
}

int main() {
    int i,j,k,z;

    for (i = 0; i < 10; i++) {
        j = i / 3;
        k = i % 3;
        z = uno(j, k);
        g1 = g1 + z;
    }
    return g1;
}
```

Esercizio 5

Considerati due vettori di nome simbolico VETT1 e VETT2 rispettivamente, aventi ciascuno 32 elementi da 8 bit, scrivere un programma in linguaggio **Assembler 8086** che faccia quanto segue:

1. memorizzi all'interno di un vettore VETT3, avente le stesse caratteristiche di VETT1 e VETT2, il risultato della somma degli elementi corrispondenti di VETT1 e VETT2;
2. stampi su un terminale connesso alla porta dello spazio di I/O avente nome simbolico TBR e registro di stato TSR, il risultato della somma espresso decimale e rappresentato tramite caratteri ASCII e seguito dai caratteri di ritorno carrello e *line feed*;
3. terminate tali operazioni ponga a 0 ogni elemento di VETT1 e di VETT2.

Esercizio 6-O

Si descriva il funzionamento di un arbitro asincrono elementare, commentando la sua tabella di stato e la sua sintesi.

Esame scritto di Calcolatori Elettronici – 31.03.2006

Corso di Laurea di Ing. Gestionale e di Ing. delle Telecomunicazioni

A.A. 2005-2006

Cognome	Nome	Matricola
---------	------	-----------

Istruzioni:

Non è ammesso l'utilizzo di materiale didattico o appunti durante questa prova, ad eccezione dell'elenco di istruzioni del processore 8086. La prova è composta di quattro esercizi: per essere ammessi all'orale è necessario completare correttamente almeno due esercizi.

Esercizio 1

Avendo a disposizione dei chip di memoria RAM del tipo 512 byte x 8, sintetizzare una memoria di tipo 1Kbyte x 16. Disegnare il montaggio commentando la soluzione scelta.

Esercizio 2

Realizzare un circuito sequenziale sincronizzato di Moore con Flip-Flop S-R che riconosca la sequenza **0,1,0,0** interallacciata.

Esercizio 3

Si considerino due vettori di nome simbolico VETT1 e VETT2 ognuno composto da 32 elementi da 8 bit. Scrivere un programma in linguaggio **Assembler 8086** che elabori il contenuto del vettore VETT1 facendo quanto segue: per ogni elemento di VETT1

1. se il bit più significativo dell'elemento vale 1, l'elemento di indice corrispondente nel vettore VETT2 viene posto uguale al valore dei quattro bit meno significativi dell'elemento considerato;
2. se il bit più significativo dell'elemento vale 0, il valore dell'elemento viene copiato nell'elemento di indice corrispondente del vettore VETT2
3. al termine della scansione del vettore VETT1, nella locazione ENNE deve essere presente il numero di volte in cui si è verificato il caso 1 e nella locazione EMME deve essere presente il numero di volte in cui si è verificato il caso 2.

Esercizio 4

Tradurre il seguente listato scritto in linguaggio C in linguaggio **Assembler 8086**, *disegnando lo stato dello stack dopo ogni chiamata di funzione*, e **commentando** opportunamente il codice scritto. Nota: i numeri interi sono rappresentati su 2 byte, i caratteri su 1 byte

```
int g;

int due(int i) {
    int k;

    k = i * 2;
    return k;
}

int uno(int x, int y) {
    int w;

    w = due(x) + y;
    return w;
}
```

```
int main() {
    int a,b;
    char c;

    a = 1;
    b = 2;
    c = 'm';

    g = uno(a, b);

    return g;
}
```

Esame scritto di Calcolatori Elettronici – 26.04.2006

Corso di Laurea di Ing. Gestionale e di Ing. delle Telecomunicazioni

A.A. 2005-2006

Cognome	Nome	Matricola
---------	------	-----------

Istruzioni: Non è ammesso l'utilizzo di materiale didattico o appunti durante questa prova, ad eccezione dell'elenco di istruzioni del processore 8086. La prova è composta di quattro esercizi: per essere ammessi all'orale è necessario completare correttamente almeno due esercizi.

Esercizio 1

Avendo a disposizione dei chip di memoria RAM del tipo 128K byte x 4, sintetizzare una memoria di tipo 512Kbyte x 8. Disegnare il montaggio commentando la soluzione scelta.

Esercizio 2

Realizzare un circuito sequenziale sincronizzato di Moore con Flip-Flop S-R che riconosca la sequenza **0,0,1,0** interallacciata.

Esercizio 3

Si considerino tre vettori di nome simbolico VETT1, VETT2, e VETT3 ognuno composto da 16 elementi da 8 bit. Scrivere un programma in linguaggio **Assembler 8086** che esegua quanto segue: per ogni elemento di VETT1

1. conti quanti bit 1 sono presenti in tale elemento e scriva tale valore nell'elemento di indice corrispondente del vettore VETT2;
2. conti quante coppie 01 sono presenti in tale elemento e scriva tale valore nell'elemento di indice corrispondente del vettore VETT3.

Esercizio 4

Tradurre il seguente listato scritto in linguaggio C in linguaggio **Assembler 8086**, *disegnando lo stato dello stack dopo ogni chiamata di funzione*, e **commentando** opportunamente il codice scritto.

Nota: i numeri interi sono rappresentati su 2 byte, i caratteri su 1 byte

```
int g;

int beta(int i) {
    int j;

    j = 10 + i;
    return j;
}

int alpha(int p, int q) {
    int r;

    r = p + q;
    return r;
}
```

```
int main() {
    int x, y;
    char c;

    x = 5;
    y = 6;
    c = 't';

    g = alpha(x, y);

    return g;
}
```

Esame scritto di Calcolatori Elettronici – 11.09.2006

Corso di Laurea di Ing. Gestionale e di Ing. delle Telecomunicazioni

A.A. 2005-2006

Cognome	Nome	Matricola
---------	------	-----------

Istruzioni: Non è ammesso l'utilizzo di materiale didattico o appunti durante questa prova, ad eccezione dell'elenco di istruzioni del processore 8086. La prova è composta di quattro esercizi: per essere ammessi all'orale è necessario completare correttamente almeno due esercizi.

Esercizio 1

Avendo a disposizione dei chip di memoria RAM del tipo 64K byte x 4, sintetizzare una memoria di tipo 128Kbyte x 8. Disegnare il montaggio commentando la soluzione scelta.

Esercizio 2

Realizzare un circuito sequenziale sincronizzato con Flip-Flop J-K che riconosca la sequenza **101** (interallacciata).

Esercizio 3

Si considerino tre vettori di nome simbolico VETT1, VETT2, e VETT3 ognuno composto da 32 elementi da 8 bit. Scrivere un programma in linguaggio **Assembler 8086** che esegua quanto segue:

1. per ogni elemento di VETT1 e VETT2 conta quanti bit sono a "1" e memorizza la loro somma all'interno dell'elemento corrispondente di VETT3;
2. quindi azzerava il contenuto di tutte le celle di VETT1 e VETT2.

Esercizio 4

Tradurre il seguente listato scritto in linguaggio C in linguaggio **Assembler 8086**, *disegnando lo stato dello stack dopo ogni chiamata di funzione*, e **commentando** opportunamente il codice scritto.

Nota: i numeri interi sono rappresentati su 2 byte, i caratteri su 1 byte

```
int g;

int delta(int i) {
    int j;

    j = 10 + i;
    return j;
}

int gamma(int p) {
    int q;

    q = p * 2;
    return q;
}
```

```
int z;

int main() {
    int a, b;

    a = 2;

    b = delta(a);
    a = gamma(a);
    z = a + b;

    return z;
}
```

Esame scritto di Calcolatori Elettronici – 25.09.2006

Corso di Laurea di Ing. Gestionale e di Ing. delle Telecomunicazioni

A.A. 2005-2006

Cognome	Nome	Matricola
---------	------	-----------

Istruzioni: Non è ammesso l'utilizzo di materiale didattico o appunti durante questa prova, ad eccezione dell'elenco di istruzioni del processore 8086. La prova è composta di quattro esercizi: per essere ammessi all'orale è necessario completare correttamente almeno due esercizi.

Esercizio 1

Avendo a disposizione un multiplexer $4 \rightarrow 1$ realizzare la funzione $z = f(A,B,C)$ indicata nella seguente tabella.

C\A,B	00	01	11	10
0	-	0	1	1
1	1	1	-	0

Esercizio 2

Realizzare un circuito sequenziale sincronizzato con Flip-Flop D+edge-triggered che riconosca le sequenze le sequenze **0,1,0,0** interallacciate.

Esercizio 3

Scrivere un programma in linguaggio **Assembler 8086** che esegua quanto segue:

1. legga dalla locazione di memoria ampia 8 bit e di nome simbolico ALFA un numero n senza segno;
2. se $n \leq 20$ calcoli lo n-esimo elemento della successione di Fibonacci, e memorizzi tale valore nella locazione di memoria ampia 16 bit di nome simbolico RISU;
3. altrimenti, memorizzi il valore $(2^{16}-1)$ nella locazione di nome simbolico RISU.

Esercizio 4

Tradurre il seguente listato scritto in linguaggio C in linguaggio **Assembler 8086**, *disegnando lo stato dello stack dopo ogni chiamata di funzione*, e **commentando** opportunamente il codice scritto.

Nota: i numeri interi sono rappresentati su 2 byte, i caratteri su 1 byte

```
int z;

int getchar(int i) {
    int base, r;

    base = 64;
    r = base + i;
    return r;
}

int nextchar(int j) {
    j = j + 1;
    return j;
}
```

```
int main() {
    int n;
    char c;

    n = 1;

    c = getchar(n);
    z = nextchar(c);

    return z;
}
```