

Politecnico di Milano - Anno Accademico 2001-2002 - Informatica B
2° Recupero – 20 Settembre 2002 – Elaborato B

NOME COGNOME	
Matricola	

REC2 /Punti	I / 2 punti	II / 2 punti	III / 1 punto	IV /1 punto	V /4 punti	REC1
Tot= / 10						

REC2 I – Calcolatore – Tradurre il semplice programma da C in Assembler, calcolare e scrivere il risultato numerico. (2 punti)

	<i>Assembler</i>	<i>Linguaggio C</i>
EXIT	EQU 2000	Void main()
IND	DW ?	#define NMAX 3
PRINTER	EQU 2	
NMAX	EQU 3	int IND;
VET	DW 1	int VET[NMAX]={1,2,3};
	DW 2	
	DW 3	
VALORE	DW ?	Int VALORE ;
INIZIO	LDA #0	{ IND = 0 ;
	STA IND	
	STA VALORE	VALORE = 0 ;
CICLO	LDA IND	while (IND != NMAX)
	SUB #NMAX	{
	JZ PRINTER	
	LDA VALORE	VALORE = VALORE + VET[IND] ;
	LDI IND	
	ADD VET(I)	
	STA VALORE	
	LDA IND	IND=IND+1;
	ADD #1	
	STA IND	
	JMP CICLO	} * end while
PRINT	LDA VALORE	Printf(VALORE) ;
	OUT PRINTER	
	JMP EXIT	} * end main
Il VALORE stampato è		6

Politecnico di Milano - Anno Accademico 2001-2002 - Informatica B
2° Recupero – 20 Settembre 2002 – Elaborato B

NOME COGNOME	
Matricola	

REC2 – II - Cosa stampa il seguente frammento di programma? (2 punti)

```
i=6;
if((i&1)==1)
{
    switch(i++)
    {
        case 6:printf("1010");
        case 7:printf("0011");
        default:printf("0001");
    }
}
else
{
    switch(++i)
    {
        case 6:printf("0101");
        case 7:printf("1100");
        default:printf("1110");
    }
}
```

OUTPUT:

11001110

REC2 - III – La rappresentazione binaria del valore 100 è **01100100**. Indicare accanto a ogni linea di questo frammento di programma l'output che essa produce. (1 punto)

```
int i=100;
if(i&1)printf("x\n");else printf("y\n");
printf("%d\n",i++);
printf("%d\n",i>>1);
```

y
100
50

Politecnico di Milano - Anno Accademico 2001-2002 - Informatica B
2° Recupero – 20 Settembre 2002 – Elaborato B

NOME COGNOME

Matricola

REC2 - IV – Riscrivi le istruzioni eseguibili del seguente frammento di codice usando solo cicli while e rispondere alla domanda sottostante (totale 1 punto)

```
int i, j, s;

s=5; i=3; j=0;
do {
    i=i+j;
    do{
        s = s-2;
    }while (j>0);
}while (s>0);
```

Soluzione

```
int i, j, s;

s=5; i=3; j=0;
i=i+j;
while (s>0)
{
    i=i+j;
    s = s-2;
    while (j>0)
    {
        s = s-2;
    }
}
```

(b) Quanto vale la variabile *s* al termine dell'esecuzione del codice? **R. -1**

REC2 - V – Files (totale 4 punti)

Il file "dati.dat" contiene delle strutture così definite:

```
typedef struct {int a; int b;} ELE;
```

Si desidera trasferire sul file "nuovi.dat" tutte le strutture il cui campo *a* è pari ed il cui campo *b* è divisibile per 7.

Scrivere il programma secondo le seguenti linee:

1. gestione file (si supponga che se dati.dat non esiste il programma termina con un messaggio di errore). (0,5 punti)
2. gestione trasferimento dati (2 punti)
3. conteggio strutture trasferite (1 punto)
4. stampa del numero delle strutture trasferite (0,5 punti)

Politecnico di Milano - Anno Accademico 2001-2002 - Informatica B
2° Recupero - 20 Settembre 2002 - Elaborato B

NOME COGNOME	
Matricola	

```
#include <stdio.h>
#include <stdlib.h>
typedef struct {int a; int b;} ELE;

main()
{FILE *fi, *fo;
  ELE e;
  int cont=0;

  carica(6);
  if((fi = fopen("dati.dat","rb")) == 0)
    {printf("errore apertura dati.dat\n");
     exit(-1);}
  if((fo = fopen("nuovi.dat","wb")) == 0)
    {printf("errore apertura nuovi.dat\n");
     exit(-2);}

  fread(&e,sizeof(ELE),1,fi);
  while(!feof(fi))
    {
      if(((e.a%2)== 0) && ((e.b %7) == 0))
        { fwrite(&e,sizeof(ELE),1,fo);
          cont++;}
      fread(&e,sizeof(ELE),1,fi);
    }
  fclose(fi); fclose(fo);
  printf("trasferiti %d valori\n",cont);
  return(0);
}
```