

Prova scritta Programmazione I FILA B - 18 Aprile 2018.

Nome e Cognome: _____

Matricola: _____

1. **4 punti** Riportare tutte le conversioni di tipo implicite; le due costanti *MAX hanno specificatore LL e U risp.. Scrivere inoltre il valore finale di b e di c, sapendo che il valore intero del carattere '2' è 50.

```
1 float f(double a) {
2   char k= '2';
3   k+= k;
4   return (a - k + 1);
5 }
6
7 int main(void) {
8   long long int a= (LLONG_MAX-
9     UINT_MAX)+1LL;
10  int b= a;
11  double c= f((b < a)? b : a);
12 }
```

VEDI CONVERSIONI FILA A

b = 1 c = -98.0

2. **3 punti** Scrivere cosa stampa il seguente programma.

```
1 int r= 6, c = 1, s, i, j;      8      c = (j>i?i:j);
2 for(i=0; i<r; i++) {          9      else
3   for(s=1; s <= r-i; s++)      10     c = (i+j+1)%i;
4   printf("-");                11   printf("-%d", c);
5                                 12 }
6   for(j=0; j <= i; j++) {      13 printf("\n");
7     if (j==0 || i==0)          14 }
```

```
-----0
-----0-0
-----0-0-1
-----0-2-0-1
-----0-2-3-0-1
-----0-2-3-4-0-1
```

3. **3 punti** Scrivere cosa stampa il seguente programma, sapendo che a si trova all'indirizzo 0x7ffec4399ffc.

```
1 int a= 0xaf, i= -03, *b= &a;
2
3 for (int* p= &i; (a== 1) ? ((*p)++, (--a)-1) : ((*p)+=2, a);
4   ++(*p)) {
5   a= (a - i);
6   printf("%d %d OK\n", a, *p);
7   if (*p > 5) { a= !a + 1; continue; } }
8   printf("%d %p %d\n", a, ((long long*) b) + 2, i);
```

```
175 -2 OK
173 0 cK
169 2 cK
163 4 cK
155 6 cK
0 0x7ffec439a00e 9
```

VEDI FILA A

4. **4 punti** Su foglio protocollo, scrivere la definizione di una funzione *matrix_multiplication* che prende in input due matrici e restituisce una terza matrice che contiene il risultato della moltiplicazione tra le prime due matrici. Per prima cosa controllare che le matrici siano moltiplicabili. *Suggerimento: crearsi la matrice risultato come int *res = (int *) malloc(rows * cols * sizeof(int)); indicizzandola quindi come un array.*
5. **3 punti** Su foglio protocollo, descrivere (anche con brevi esempi di codice) le differenti zone di memoria dove possono essere memorizzate le variabili in C, e le loro caratteristiche.
6. **2 punti** Su foglio protocollo, scrivere due file, *main.c* e *write.c*: in *main.c* viene stampato una variabile a (di tipo *int*) definita in *write.c*, mentre in *write.c* viene stampata da una funzione di nome *write* una variabile b (di tipo *int*) definita in *main.c*. Scrivere i due file in modo che possano essere compilati anche separatamente.

VEDI FILA A

7. **4 punti** Scrivere cosa stampa il seguente programma, sapendo che le variabili a , b , c in *main* sono memorizzate rispettivamente agli indirizzi $0x7ffeb7d2964$, $0x7ffeb7d2970$ e $0x7ffeb7d2974$.

```

1 static int a= 4;
2 int f(int* x, int y, int* z) {
3     int res= 2;
4     if (--a > 0) {
5         a ? res= (y+=2) + (*z)++, (y+= *x) : 3;
6         printf("%d %d %p %d\n", *x, y, z+1, res);
7         res+= f(&y, *x, z);
8         return (res);
9     }
10    else
11        return res;
12 }
13
14 int main(void) {
15     int a= 3, b= 5, c= 3;
16     b= f(&b, a, &c);
17     printf("%d %d %d\n", a, b, c);
18 }

```

0x7ffeb7d2978
0x7ffeb7d2978
0x7ffeb7d2978

Altri valori uguali a FILA A: cambia solo il valore dell'indirizzo di memoria

8. **3 punti** Data la seguente *struct Node*, definire una funzione ricorsiva di nome *recursive_print* che stampa il campo *info* a partire dall'ultimo elemento di una lista di tali nodi. La funzione ha come parametro solamente una variabile di tipo *struct Node**.

```

1 struct Node {
2     int info= 0;
3     struct Node* pNext= NULL;
4 }
5

```

VEDI FILA A

9. **4 punti** Cerchiare le affermazioni vere dato $\text{long long } a[3] = \{129 + 256, \text{INT_MAX}, \text{LLONG_MIN}\};$
 $\text{short int } *p = (\text{short}*) a; \text{char } *q = (\text{char}*) a; p[1] = 8192, p[3] = 8190, *(q+15) = 72, p[9] = 8192*2+1;$
 sapendo che i tre tipi usati occupano 8, 2, e 1 byte, e $8192 = 2^{13}$ (valori rappresentati in *little endian* e complemento a due). Scrivere la mappa di memoria e giustificare le affermazioni (vere o false). Gli operatori $|$ e $\&$ ritornano rispettivamente l'or e l'and bit-a-bit dei due operandi. Supporre che i tipi *char* siano tipi con segno: il bit più significativo uguale a 1 rappresenta un valore negativo.

A. $(q[11] | q[23]) + q[1]$ B. $((int)(p + 10) - (int)(a + 1)) + q[18] \% 13$ C. $q[23] - q[0] + q[18]$
 D. $(*p + 5) - p[3] \% 2$ E. $(\&a[3] - (a + 1)) + q[1] \leq 4$ F. $(*p + 4) \& *(p + 5) == p[5]$

10. **1 punto** Quale comando di compilazione permette di ottenere anche la traduzione in linguaggio assembly del file *file.c*?

☐ gcc -A file.c; ☒ gcc -S file.c; ☐ gcc -assembler file.c; ☒ gcc -S -c file.c; ☐ gdb file.c;
☐ gcc -E file.c