

Prova scritta Programmazione I - 17 Aprile 2019 - FILA B

Nome e Cognome: _____

Matricola: _____

1. **3 punti** Riportare le conversioni di tipo implicite e scrivere quanto valgono alla fine le variabili x , k , z , sapendo che `UINT_MAX` vale 4294967295 (le conversioni da `*_MAX` non comportano conversioni).

```
1 unsigned int x = UINT_MAX, k= 10U, z
   = -3;
2 int a= -2.3f;
3 long long y = LLONG_MAX;
4 long j= 2LL;
5 x = y - x;
6 k = a < k;
```

conversioni come fila A

`z == (UINT_MAX +1) - 3 == 4294967293`

i valori di k e z sono 0 come in fila A

2. **4 punti** Su foglio protocollo, scrivere un programma che prende da input un valore intero n , e stampa tutti i numeri primi tra 0 e n : per esempio, se $n == 14$, stampare 2, 3, 5, 7, 11, 13. Stampare come in figura.

```
2
3 5
7 11 13
17 19 23 29
31 37 41 43 47
```

Vedere fila A

3. **3 punti** Scrivere cosa stampa il seguente programma, sapendo che b si trova all'indirizzo 0x1047e601d.

```
1 int a = 0x19, b= 017;
2
3 void f(void) {
4   while (b > 7? b-=2 : b) {b--; ++a;} }
5
6 int main() {
7   printf("%p\n", &b);
8   for (int i= 1 ; f(), b= b+i; i++, printf("%d
9     %d\n", a, b))
10     if (i > 4) break;
11   else do {(&a)++;} while (!a && a++);
12   printf("%p", ((long long*) &b) +1);
13 }
```

Vedere fila A

**cambiano solo i puntatori stampati
0x1047e601d
e 0x1047e6025**

4. **4 punti** Data la seguente struttura che rappresenta ogni nodo di una lista, definire una funzione di nome *duplicate* che prende come parametro *int infoToDuplicate* ed il puntatore al puntatore di una lista (cioè ad un *Node**). La funzione aggiunge un nuovo elemento con valore *infoToDuplicate* in testa alla lista solo se la lista contiene già un elemento con valore identico a *infoToDuplicate*.

```
1 struct Node {
2   int info;
3   struct Node* pNext;
4 }
```

Vedere fila A

5. **3 punti** Su foglio protocollo, scrivere una funzione ricorsiva di nome *palindrome* che controlla se una stringa, passata come parametro alla funzione è palindroma oppure no (*return* 1 oppure 0). Non utilizzare variabili *static* o globali. Per esempio, *itopinonavevanonipoti* è palindroma (destra-sinistra = sinistra-destra).

6. 3 punti Su foglio protocollo, scrivere il contenuto di due file, *main.c* e *secondary.c*. I due file devono essere compilati insieme producendo con successo l'eseguibile di nome *myexe*: scrivere il comando per fare ciò. I due file devono poter essere compilati anche indipendentemente per produrre un file oggetto; scrivere il comando per fare ciò. In *main.c* ci devono essere esattamente un tentativo di definizione che rimane dichiarazione, una definizione di variabile e una di funzione entrambi con *external linkage*, una dichiarazione di funzione con *external linkage*. In *secondary.c* ci devono essere esattamente una definizione di funzione con *external linkage*, una variabile con *internal linkage* e due variabili con *no linkage*.

Vedere fila A

7. 4 punti Scrivere cosa stampa il seguente programma.

```

1 int a, r= 2;
2
3 int f(int* y, int *x, int z) {
4     extern int a;
5     a = a + 1;
6     if (a < 4) {
7         r += ++(*x) + ++(*y), (*y+= z);
8         r= r + f(x, y, z);
9         printf("%d %d %d %d\n", *x, *y, z, r);
10        return (r/a);
11        printf("%d\n", r++); }
12    else
13        return r/a; }
14
15 int main(void) {
16     int a= 3, b= 2, c= 5;
17     b= f(&b, &c, a);
18     printf("%d %d %d\n", a, b, c); }
```

Vedere fila A

8. 4 punti Cerchiare le affermazioni vere dato $int\ a[5] = \{INT_MAX - 7, 1287, INT_MIN + 528, -10, 312\};$ $short\ int\ *p = (short\ int*)\ a;$ $char\ *q = (char*)\ a;$ $p[3] = SHRT_MAX,$ $p[5] += 2048,$ $q[18] = \sim q[19];$ sapendo che i tre tipi usati occupano 4, 2, e 1 byte, con valori rappresentati in *little endian* e complemento a due. Scrivere la mappa di memoria e giustificare le affermazioni (vere o false).

- ☐ A. $\sim (\&p[5] - \&p[1] - q[9] * 2)$ B. $(q[0] | q[1]) + q[17]$ ☒ C. $((int)(q + 13) - (int)(a + 1)) - q[1] \% 3$
☒ D. $q[9] + q[11] > -120$ E. $(*(p + 7) \& p[9]) \% 5$ F. $*((short*) \& q[17]) > 0$

Vedere fila A per mappa di memoria, esercizio 9 e esercizio 10

9. 2 punti Quali affermazioni sono vere?

☐ le espressioni contengono comandi; ☐ $i++$; è una espressione; ☐ *break* è un comando. ☐ $a = 4$ è una espressione; ☐ *return* ha come argomento un comando. ☐ i comandi sono separati da virgole.

10. 1 punti Data l'espressione $b = a + + + c + +$ riportare qui il numero di effetti collaterali presenti: