# Informatica A

**Cognome (IN STAMPATELLO)**

**Nome**

**Matricola o Codice Persona**

Istruzioni

1. Non separate questi fogli. Scrivete la soluzione **solo** **sui fogli distribuiti**, utilizzando il retro delle pagine in caso di necessità.
2. **È possibile scrivere a matita** (e non ricalcare al momento della consegna).
3. Ogni parte non cancellata sarà considerata parte integrante della soluzione.
4. È **vietato** utilizzare **calcolatrici** o **telefoni**. Chi tenti di farlo vedrà **annullata** la sua prova.
5. Non è ammesso consultare **libri** o **appunti** o altro.
6. Qualsiasi **tentativo** di comunicare con altri studenti comporta **l’espulsione** dall’aula.
7. È possibile **ritirarsi senza penalità** lasciando il tema d’esame con nome e cognome.
8. Non è possibile lasciare l’aula conservando il tema della prova in corso.
9. Tempo a disposizione: 2 ore e 30 minuti

### Valoredegli esercizi, voti parziali e voto finale:

**Esercizio 1 ( 2 punti )**

**Esercizio 2 ( 4 punti )**

**Esercizio 3 ( 4 punti )**

**Esercizio 4 ( 8 punti )**

**Esercizio 5 (10 punti )**

**Esercizio 6 ( 4 punti )**

**Totale: ( 32 punti )**

## Esercizio 1 - Algebra di Boole, Aritmetica Binaria, Codifica delle Informazioni (2 punti)

1. Si costruisca la tabella di verità della seguente espressione booleana, badando alla precedenza tra gli operatori logici. (1 punto).

**( A or ( B and ( not C ) ) ) and ( A and B and C )**

1. Si stabilisca il minimo numero di bit sufficiente a rappresentare in complemento a due entrambi i numeri A = 19dec e B = –95dec

Si converta A e B in CP2 usando bit

Si calcoli quindi (A+B) e (A–B) in complemento a due e si indichi se si genera riporto sulla colonna dei bit più significativi e se si verifica overflow (1 punto).

## Esercizio 2 ( 4 punti )

Lo schema sottostante descrive il processo di pubblicazione di articoli scientifici. Ogni articolo è scritto da uno o più autori (ricercatori) e viene sottomesso ad una rivista per essere pubblicato. Ogni articolo viene recensito da tre reviewer (anch’essi ricercatori), e nel caso almeno due di essi propendano per l’accettazione, l’articolo viene accettato per la pubblicazione.

RICERCATORE (CF, Nome, Cognome, Università)

ARTICOLO (Titolo, Data\_Sottomissione, Rivista, Accettato)

ARTICOLO\_AUTORE (Titolo, CF\_Autore)

REVIEW (Titolo, CF\_Reviewer, Da\_Accettare)

Scrivere una query che estrae cognome e nome dei ricercatori che non hanno mai sottomesso articoli ad una rivista per la quale sono stati anche reviewer.

RICERCATORE (CF, Nome, Cognome, Università)

ARTICOLO (Titolo, Data\_Sottomissione, Rivista, Accettato)

ARTICOLO\_AUTORE (Titolo, CF\_Autore)

REVIEW (Titolo, CF\_Reviewer, Da\_Accettare)

Scrivere una query che estrae per ogni rivista cognome e nome dell’autore con più articoli sottomessi.

**Esercizio 3** ( 4 punti )

Si dica cosa stampa il seguente codice

**#include <stdio.h>**

**#include <stdlib.h>**

**typedef struct el {**

**int intero;**

**char c;**

**struct el \* next; } nodo;**

**void f(char \* str, nodo \*\* n, int i) {**

**if( i >= 4 )**

**return;**

**if(i==2)**

**(\*(\*n+i)).c = 'A';**

**else**

**(\*(\*n+i)).c = str[0] + (\*n+i)->intero;**

**printf("%c", (\*n+i)->c );**

**f( str+1, n, i+1 );**

**}**

**void foo( nodo nodes[], int i ) {**

**int j = 1;**

**while( i-- > 0 ){**

**(\*(nodes)).intero = j;**

**nodes++;**

**j++;**

**}**

**}**

**int main() {**

**char str[] = "AFAJ";**

**int dim,i;**

**for(i=0;i<4;i++)**

**str[i]=str[i]+1;**

**dim = sizeof(nodo);**

**nodo \* a = (nodo \*)malloc( 4 \* dim );**

**foo( a, 4 );**

**f( str, &a, 0 );**

**return 0;**

**}**

## Esercizio 4 ( 8 punti )

Definiamo percorso in una matrice di caratteri una sequenza di passi che partono dalla posizione (0,0) e finiscono nella posizione (N-1, N-1) effettuando solo movimenti dall’alto verso il basso o da sinistra verso destra (nessun altro movimento è consentito). Ad ogni percorso è possibile associare la stringa di caratteri che si ottiene concatenando i caratteri attraversati dal percorso stesso.

Definiamo percorso palindromo un percorso a cui è associata una stringa palindroma.

Si scriva una funzione percorsiPalindromi che ricevuta una matrice di caratteri NxN restituisce il numero di percorsi palindromi.

## Esercizio 5 ( 10 punti )

Scrivere una funzione che, data una lista i cui elementi sono liste ordinate di interi, elimina le sottoliste che contengono numeri ripetuti.

Esercizio 6 **( 4 punti )**

Si consideri la seguente definizione di un albero binario:

typedef struct nodeS {

int v;

struct nodeS \* left, right;

} node;

typedef node \* tree;

Scrivere una funzione che, preso in input un albero binario, calcola la somma pesata dei valori presenti nell’albero, dove il peso di ogni valore è uguale al livello nell’albero in cui si trova (assumendo che la radice è al livello 1).