

## Prova pratica

### Primo tema

1.

Il candidato implementi un algoritmo per il rilevamento dei bordi agendo direttamente sui pixel, senza usare edge detector forniti da librerie o toolbox Matlab. Il candidato elabori poi l'immagine fornita per estrarre i bordi, salvando il risultato nella cartella messa a disposizione (**punti 13**).

NOTA: qualora il candidato optasse per l'utilizzo delle librerie OpenCV, troverà gli header nella directory `/usr/include/opencv2`, e i file di libreria in un percorso standard (che non è necessario specificare al compilatore).

2.

Il candidato implementi il seguente algoritmo di pianificazione del moto per robot mobili in ambienti di cui è disponibile una mappa di occupazione. Il valore di ogni cella della mappa di occupazione sia: '-1' se l'elemento è occupato da un ostacolo, oppure un numero naturale che rappresenta il numero di mosse necessario a raggiungere il GOAL (rappresentato dal valore 0) a partire dalla posizione dell'elemento.

Dato il file di testo fornito <occupancy grid map 1.txt> che rappresenta la mappa di occupazione di un ambiente, il candidato implementi il wavefront propagation algorithm in un linguaggio di programmazione a sua scelta. L'output deve essere un file di testo simile a quello rappresentato in figura (cioè una matrice di numeri, ma in modalità testo, non in modalità grafica). L'output deve essere salvato nella cartella messa a disposizione (**punti 17**).

Inoltre, per entrambi gli esercizi il candidato illustri brevemente il procedimento seguito sui fogli protocollo messi a sua disposizione.

#### WAVEFRONT PROPAGATION ALGORITHM

1. Initialize  $W_0 = X_G$ ;  $i = 0$ .
2. Initialize  $W_{i+1} = \emptyset$ .
3. For every  $x \in W_i$ , assign  $\phi(x) = i$  and insert all unexplored neighbors of  $x$  into  $W_{i+1}$ .
4. If  $W_{i+1} = \emptyset$ , then terminate; otherwise, let  $i := i + 1$  and go to Step 2.

Figura 1. Pseudocodice del wavefront algorithm.