Surname, Name, ID number: ____________________

# Autonomous and Mobile Robotics M

22 December 2023 - Theory

*Some questions may have more than one correct answers: for each question, indicate all the correct answers.*

1. Given the constraints matrix equation in Pfaffian form $A(q)\dot{q} = 0$, the admissible robot speed:
   - ⊗ is generated by a matrix $G(q)$ such that $\mathrm{Im}(G(q)) = \mathrm{Ker}(A(q)), \forall q$
   - ○ is generated by a matrix $G(q)$ such that $\mathrm{Ker}(G(q)) = \mathrm{Im}(A(q)), \forall q$
   - ○ is generated by a matrix $G(q)$ such that $G(q) = A(q)^T, \forall q$
2. For a unicycle robot, given the geometric trajectory $x(s)$, $y(s)$, $\theta(s)$, it is possible to write the steering input $\omega(s)$ as:
   - ○ $\omega(s) = (\theta''(s)x'(s) - \theta''(s)y'(s))/(x'(s)^2 + y'(s)^2)$
   - ⊗ $\omega(s) = (y''(s)x'(s) - x''(s)y'(s))/(x'(s)^2 + y'(s)^2)$
   - ○ $\omega(s) = (y''(s)\theta'(s) - x''(s)\theta'(s))/(x'(s)^2 - y'(s)^2)$
3. Consider Odometry for WMR:
   - ⊗ it represents a reliable estimation of the robot position over a single evaluation step
   - ○ it presents an exact estimation for the $x$ and $y$ variables if the precise reconstruction method is used
   - ○ the precise reconstruction method is not affected by changes of the steering angle overa a single step
4. Examples of map-based navigation algorithms are:
   - ⊗ distance transform planning;
   - ⊗ A$^*$ and D$^*$;
   - ○ bug algorithms.
5. A process satisfies the Markov property if:
   - ○ the agent state is the same as the environment state
   - ⊗ one can make predictions for the future of the process based solely on its present state
   - ○ one can make predictions for the future of the process only based on the process full history
6. The Bellman optimality equation for the state value function can be written as
   - ○ $v_*(s) = \max v_\pi(s)$
   - ⊗ $v_*(s) = \max_{a\in\mathcal{A}} q_*(s,a)$
   - ⊗ $v_*(s) = \max_{a\in\mathcal{A}} \mathbb{E}_*[G_t|S_t = s, A_t = a]$
7. The relative probability of the trajectory obtained following a target policy $\pi$ w.r.t. the behavior policy $\mu$ is:
   - ○ $\rho_t^T = \prod_{k=t}^{T-1} \pi(A_k|S_k) \prod_{k=t}^{T-1} \mu(A_k|S_k)$
   - ⊗ $\rho_t^T = \prod_{k=t}^{T-1} \pi(A_k|S_k) / \prod_{k=t}^{T-1} \mu(A_k|S_k)$
   - ○ $\rho_t^T = \prod_{k=t}^{T-1} \mu(A_k|S_k) / \prod_{k=t}^{T-1} \pi(A_k|S_k)$
8. The $\lambda$-return is defined as:
   - ○ $G_t^\lambda = \sum_{k=0}^{\infty} \lambda^k R_{t+k+1}$
   - ○ $G_t^\lambda = R_{t+1} + \lambda V(S_{t+1})$
   - ⊗ $G_t^\lambda = (1-\lambda)\sum_{n=1}^{\infty} \lambda^{n-1} G_t^{(n)}$
9. In value function approximation by stochastic gradient descent, the parameter vector update is defined as:
   - ○ $\Delta\mathbf{w} = -\frac{1}{2}\alpha\nabla_{\mathbf{w}} J(\mathbf{w})$
   - ⊗ $\Delta\mathbf{w} = \alpha\,(v_\pi(S) - \hat{v}(S,\mathbf{w}))\nabla_{\mathbf{w}}\hat{v}(S,\mathbf{w})$
   - ○ $\Delta\mathbf{w} = \alpha\,\mathbb{E}_\pi\left[(v_\pi(S) - \hat{v}(S,\mathbf{w}))\nabla_{\mathbf{w}}\hat{v}(S,\mathbf{w})\right]$
10. Given two tasks $a \in^{m_a}$ and $b \in^{m_b}$ which Jacobian matrices with respect to the robot configuration are $J_a$ and $J_b$, $a$ and $b$ are said:
    - ⊗ *orthogonal* if $J_a J_b^\# = 0_{m_a\times m_b}$, where $^\#$ represents the matrix pseudoinverse;
    - ⊗ *dependent* if $\mathrm{rank}(J_a^T) + \mathrm{rank}(J_b^T) > \mathrm{rank}([J_a^T\ J_b^T])$;
    - ○ *independent* if $\mathrm{rank}(J_a^T) + \mathrm{rank}(J_b^T) < \mathrm{rank}([J_a^T\ J_b^T])$;

# Autonomous and Mobile Robotics M

22 December 2023 - Exercise

*The student is asked to solve the following problem.*

Let us consider a fully observable environment with 5 states $s_{\{1,...,5\}}$.

| $s_1$ | $s_2$ | $s_3$ | $s_4$ | $s_5$ |
|---|---|---|---|---|

- Action set : {TryLeft, TryRight}
- Rewards:
  - +1 in state $s_1$
  - -1 in state $s_3$
  - +2 in state $s_5$
  - 0 in all other states
- Transition model:
  - $p(s_1|s_1, \text{TryLeft}) = p(s_5|s_5, \text{TryRight}) = 1$
  - $p(s_1|s_1, \text{TryRight}) = p(s_2|s_1, \text{TryRight}) = 0.5$
  - $p(s_1|s_2, \text{TryLeft}) = p(s_2|s_2, \text{TryLeft}) = 0.5$
  - $p(s_2|s_2, \text{TryRight}) = p(s_3|s_2, \text{TryRight}) = 0.5$
  - ...
- Policy: $\pi(\text{TryLeft}|s_{\{1,...,5\}}) = \pi(\text{TryRight}|s_{\{1,...,5\}}) = 0.5$
- Discount factor $\gamma = 1$

Starting from an arbitrary initialisation of the state value function, compute the first iteration of the state value function evaluation provided by a Dynamic Programming algorithm with asyncronous backup assuming the random policy $\pi$.

| $v_\pi(s_1)$ | $v_\pi(s_2)$ | $v_\pi(s_3)$ | $v_\pi(s_4)$ | $v_\pi(s_5)$ |
|---|---|---|---|---|
| | | | | |

**Solution:**

The state value function is initialized to 0 for all the states.

| $v_\pi(s_1)$ | $v_\pi(s_2)$ | $v_\pi(s_3)$ | $v_\pi(s_4)$ | $v_\pi(s_5)$ |
|---|---|---|---|---|
| 0.75 | 0.1875 | -0.4531 | 0.1367 | 1.5342 |