# Lab 3 Task 6 Solution

## 6) Implement class RandomPlayer

```python
import random

class Player:
    """A player in number game

    === Attributes ===
    name:
        The name of player

    === Representation invariants ===
    - len(name.strip()) != 0
    - 0 <= self.current <= self.goal
    - 0 < self.min_step <= self.max_step <= self.goal
    """
    name: str

    def __init__(self, name: str) -> None:
        """Initialize this Player

            Precondition:
                - len(name.strip()) != 0
        """
        self.name = name

    def move(self, current: int, min_step: int,
             max_step: int, goal: int) -> int:
        """Return amount of steps taken by a player

            Precondition:
                - 0 < min_step <= max_step <= goal
                - 0 <= self.current <= self.goal
        """
        raise NotImplementedError

class RandomPlayer(Player):

    def move(self, current: int, min_step: int,
             max_step: int, goal: int) -> int:
        return random.randint(min_step, max_step)
```

Listing 1: task_6_solution.py