**Relatório de Programação Orientada para Objetos**

Leia atentamente e siga as intruções. Substitua abaixo XX pelo número do grupo e <NOME e NÚMERO> pelos nomes e números dos elementos do grupo.

# Grupo XX:

## José Cabral 110864

## Sérgio Silva 111052

# Ao entregar este trabalho CADA um dos elementos do grupo declara que:

1. Reconhece que existe plágio sempre que utiliza trabalho de outros e pretende ser seu
2. A resposta a este trabalho foi elaborada pelos elementos indicados acima (mesmo que com ajuda de ferramentas de apoio à programação) e ambos conhecem integralmente o funcionamento do trabalho associado a este relatório
3. Não partilhou o código com outros colegas
4. Não utilizou o código desenvolvido por terceiros (exceto em bibliotecas standard e quando explicitamente referido) nem código partilhado por outros colegas

# Funções de cada elemento do grupo:

Indique com “**X**” as uma a três das funções mais relevantes para cada membro do grupo.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Número de aluno | IDEALIZAÇÃO | codificação | Apoio à Codificação | testes |
| 110864 |  | X | X | X |
| 111052 |  | X | X | X |

# CHECKLIST:

Preencha a coluna com A (totalmente cumprido), B (quase totalmente cumprido), C (parcialmente cumprido), D (com lacunas) ou E( não cumprido)

|  |  |
| --- | --- |
| Grau de cumprimento | Requisitos |
| A | Ler o ficheiro de configuração e representar os elementos na GUI |
| A | Implementar o movimento da empilhadora. |
| A | Implementar a interação da empilhadora com os caixotes |
| A | Implementar a interação da empilhadora com outros objetos |
| B | Modularização, distribuição, encapsulamento e legibilidade do código; |
| A | Definição e utilização de uma hierarquia adequada de herança; |
| A | Definição e utilização correta de interfaces; |
| B | Originalidade e extras |

Se indicou A ou B no ultimo item, diga quais os extras implementados na secção seguinte (Discussão de opções tomadas … ).

# Discussão de OPções Tomadas e/OU Funcionamento além do enunciado

**Se achar necessário** descreva aqui opções tomadas que podem ser críticas para a interpretação da sua implementação ou funcionamento do protótipo além do que foi indicado no enunciado.

Os extras mencionados na secção anteriori podem ser o uso de comparadores, predicados, padrões de desenho, expressões lambda, coleções, exceções, ou outras particularidades que mereçam ser mencionadas.

Aspetos positivos e originais da sua implementação que mereçam ser destacados

Caso não tenha nada a indicar apague esta secção.

--------------------------------------------------------------------------------------------------------------------------------------------------------

Escolhemos implementar uma interface “link” que descreve todas as interações entre todos os GameElements pois achámos que simplificaria a compreensão do código, e se quiséssemos implementar novos GameElements, este processo seria facilitado caso quiséssemos implementar interações

A meio do trabalho, reparámos que o excessivo de loops tornava o jogo extremamente lento, e o nosso código não estava nada otimizado, mas pelo menos funcionava correctamente e estava organizado. Decidimos proceder com a otimização do código, reduzindo o tamanho e a ocorrência dos loops, e implementámos heranças entre o Caixote e a Palete, e entre a Parede e a Parede Rachada. O resultado satisfez-nos bastante

Como "extras", implementámos um Comparador para organizar os scores, e fizemos com que fosse possível recomeçar o nível atual se o jogador clicar na tecla "R". Também lançámos algumas excessões para problemas que possam ocorrer na leitura de ficheiros.

Para implementar o Score optamos por criar duas pastas uma chamada “scores” e outra “Top3” na pasta “scores” sao guardados o historico de todos os scores de cada nivel,na pasta “Top3” sao registados os top 3 de cada nivel ,ordenado consoante aquele que gastou menos bateria para ganhar o nivel.

Feliz natal e Bom ano!!!