

Datei-Struktur und Aufgabe 6

Samstag, 24. September 2022 10:16

Datei-Struktur des Portfolios

1a	/assets/map map.h / map.cpp
1b	/assets/map map.h / map.cpp
1c	item_base.h/ item_base.cpp Kindklassen (item_chest, item_potion, etc.)
2a	inventory_template.h inventory_ui.h / inventory_ui.cpp item_ring.h / item_ring.cpp item_dagger.h / item_dagger.cpp item_armor.h / item_armor.cpp
2b	character_base.h / character_base.cpp character_player.h / character_player.cpp
3a	character_player.h / character_player.cpp
3b	character_player.h / character_player.cpp (M um mehrere Items ins Player Inventar zu legen, TAB und ENTER um die Sortierfunktion auszuwählen, SPACE um die Attribute eines Items zu sehen, N um das Inventar wieder zu leeren)
4a	map.h / map.cpp character_base.h / character_base.cpp character_player.h / character_player.cpp character_robot.h / character_robot.cpp
5a	alle src-Dateien und assets
6a und b	dieses PDF Dokument

Aufgabe 6

a)

Algorithmus	Kosten	Anzahl
void selSort(int array[], in size) {	C1	1
for (int step = 0; step < size - 1; step++) { //closes c7	C2	n
int min_idx = step;	C3	n-1
for (int i = step + 1; i < size; i++) { //closes c6	C4	$\sum_1^n + n-1$
if (array[i] < array[min_idx]){	C5	$\sum_1^n + n-1$
min_idx = i; } }	C6	$\sum_1^n + n-1 - 1$
swap(&array[min_idx], &array[step]); }	C7	n-1

b)

$$O(n) = (3 \cdot (n-1)) + (2 \cdot \sum_1^n + (n-1)) + (\sum_1^n + (n-1) - 1) + 1$$

$$O(n) = (3 \cdot (n-1)) + (3 \cdot \sum_1^n + (n-1))$$

$$\underline{O(n) = (3 \cdot (n-1)) + (3 \cdot \sum_1^n + (n-1))}$$