

S1_ImP-Prüfungsleistung

Grundaufgabe

Simulation einer Paketstation über 2 Wochen hinweg

- Es sollen pro Echtzeitsekunde 10 Minuten in der Simulation vergehen

1-4 Uhr sollen jedoch in einer Echtzeitsekunde vergehen

- Zusätzlich hierzu soll es einen "Turbo-Knopf" geben, der die Geschwindigkeit auf 100 Minuten erhöht
- Montag bis Freitag Entleerungen 10:30 und 18:30
- Samstag Entleerungen 14:00
- Sonntags KEINE Entleerung
- Entleerung selbst dauert 10 Minuten (also eine Echtzeitsekunde)

-> Innerhalb dieser Zeit können die Kunden die Paketstation NICHT benutzen!

-
- Pro Echtzeitsekunde können 2 bis 10 Paket-Interaktionen geschehen

-> Median sind 5 Pakete

- Pakete sollen ihrer entsprechenden Größe ein Fach bekommen
- Ist die Größe vollständig belegt, wird das nächstgrößere Fach belegt
- Sind alle passenden belegt, wird der Kunde weggeschickt
- Kunden sind bereit bis zu 20 Minuten warten, wenn ein Entleerungstermin bevor steht.

-> die Entleerungszeit wird dann auch noch gewartet wenn das der Fall ist

- Mitarbeiter führt erst die Entleerung aus, dann befüllt er
- Befüllt genau so wie die Kunden
- Wenn die Paketstation voll ist, nimmt der Mitarbeiter die zu befüllenden Pakete für das nächste Mal mit
- Pakete werden innerhalb von 4 Werktagen entweder abgeholt oder durch Mitarbeiter zurückgesendet
- Möglichkeit, dass Kunden an Kunden, die dieselbe Station benutzen versenden

-> in diesem Fall muss aber erst ein Mitarbeiter kommen, danach ist das Paket zur Abholung geflaggt

- 250 Kunden
- Durchschnittlich erhält jeder Kunde 2 PAKETE PRO WOCHE

Trafficzeiten:

Zeit	Traffic-Modifikator
0	1%
1-4	1%
5	1%

Zeit	Traffic-Modifikator
6	15%
7	100%
8	150%
9	100%
10	50%
11	50%
12	100%
13	100%
14	50%
15	50%
16	100%
17	150%
18	100%
19	50%
20	50%
21	15%
22	1%
23	1%

Fächer:

Fachgröße	Anzahl
XS	4
S	5
M	21
L	10
XL	2

- Pseudo-Graphische Konsolenausgabe

Dokumentation

Headers

customer_ops.h

DEPRECATED, READY FOR DELETION

debug.h

Diese Header-Datei ist ausschließlich fürs debuggen vorgesehen gewesen. Die debug_output-Funktion gibt, wenn man sie aufruft alle Variablen mit ihren aktuellen Werten aus. Mit dieser Funktion konnten wir Fehler überprüfen und nachvollziehen, was das Programm tatsächlich macht.

Diese Header-Datei hat KEINERLEI Funktion oder Bedeutung im regulären Einsatz des Programms!

DVL.h

Diese Header-Datei ist für die Initialisierung und Bearbeitung unserer Transaktions-Liste Diese Liste, weiter unten auch noch einmal in dem Abschnitt über main.c, beinhaltet alle geplanten Transaktionen (Paketlieferungen), die getätigt werden. Mit den Funktionen in dieser Header-Datei können geplante Transaktionen der Liste hinzugefügt und auch entfernt werden

helperFunc.h

In dieser Header-Datei wurden alle Helfer-Funktionen ausgelagert. Funktionen, die nicht zwingend einer anderen Header-Datei zugewiesen werden konnten und generell auch nur als Hilfe, z.B. zur Umwandlung der Zeiteinheit vom internen Format (welches Sekunden sind, die stetig hochzählen) zu einem Ausgabeformat (also Stunde, Minute, Tag)

output.h

DEPRECATED, READY FOR DELETION

package_ops.h

In dieser Header-Datei befinden sich alle Funktionen, die die Transaktionen an der Paketstation selbst darstellen.

Die Funktion input_packages schaut sich bei Aufruf die Größe des Paketes an und fängt dann an die Paketstation nach passenden, leeren Fächern abzusuchen. Findet die Funktion kein exakt der größe passendes Fach, wird die nächst größere Größe belegt (S ist vollständig belegt -> suche bei M weiter).

Die Funktion output_packages schaut sich bei Aufruf die IDs der Pakete innerhalb der Paketstation an. Findet die Funktion ein Paket mit der selben ID, wie beim Aufruf mitgegeben wurde (also die ID des Kundens an der Station), wird das Paket entnommen. Falls mehrere Pakete mit der selben ID existieren, werden alle entnommen. Im Falle, dass der Mitarbeiter der Paketstation (mit der besonderen ID 300), die Funktion aufruft, wird zusätzlich noch bei allen Paketen ein Flag gesetzt, dass diese, falls sie intern sind, jetzt abholbar seien.

Die Funktion age_packages inkrementiert jeden Tag das Alter des Paketes. Der Mitarbeiter wird bei Aufruf der output_packages alle Pakete mit einer age-Variable ≥ 4 entfernen, egal ob sie seiner ID entsprechen oder nicht.

structs.h

In dieser Header-Datei befinden sich die Initialisierungen aller structs, die innerhalb des Programms genutzt werden. Hier kann man die Datenstruktur am Besten einsehen.

worker.h

DEPRECATED, READY FOR DELETION

main.c

In der Haupt-Datei befinden sich die folgenden Funktionen:

Initialize: Diese Funktion wird am Anfang des Programms einmal aufgerufen und setzt alle Anfangs-Einstellungen, wie z.B. Die Anzahl an Postfächern, die grafische Darstellung usw.

getCharPosition: Diese Funktion sucht das "#"

refreshScreen: Diese Funktion erneuert die grafische Ausgabe

writeToTextBuffer: Diese Funktion schreibt das gegebene in den Buffer für späteren Nutzen

drawScreen: Diese Funktion bereitet die grafische Ausgabe vor

readIn: Diese Funktion liest eine TXT-Datei aus, die eine Vorlage für die grafische Ausgabe beinhaltet

calculateTimeStep: Diese Funktion berechnet alles, was für einen TimeStep notwendig ist, darunter auch die zu liefernden Pakete über generatePackage() und die Kunden, die Pakete abholen/abgeben wollen über queueCustomer().

generatePackage: Diese Funktion generiert Pakete und packt sie in die transactionQueue

queueCustomers: Diese Funktion fügt Kunden zur customerQueue hinzu, die dann abgearbeitet wird um die Pakete abzuholen/abzugeben

dequeueCustomers: Diese Funktion entfernt Kunden aus der customerQueue, wenn es abgegeben/abgeholt wurde