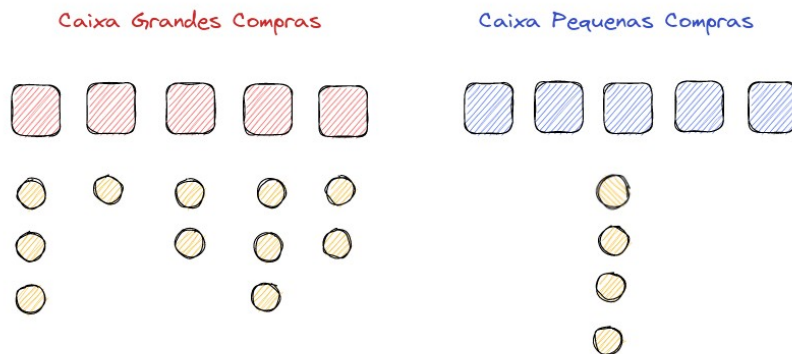


MP2 – 2022.2

Em supermercados existem 2 tipos de caixas: O caixa de grandes compras, para mais de 20 produtos (CaixaG), e o caixa para pequenas compras, até 20 produtos (CaixaP). Sabe-se que, em um supermercado como o representado no esquema abaixo, há 5 caixas do tipo CaixaG, que possuem uma fila individual cada, e 5 caixas do tipo CaixaP, que dividem uma única fila com capacidade para 100k pessoas. Escreva um programa em C para determinar que tipo de modelo de fila é mais efetivo atendendo clientes durante um dia.



Seu código deverá funcionar da seguinte forma:

1. **Sortear um número** real N entre 0 e 1, de tal forma que, se $N \leq 0.5$, nenhum novo cliente chegou. Senão, um novo cliente chegou. Neste caso, você deve
 - a) Sortear um inteiro entre 1 e 40 que representa a quantidade de itens no carrinho desse cliente;
 - b) Sortear um valor real entre 0 e 1 que representa o tempo de atendimento desse cliente;
2. **Inserir os clientes** na fila P ou G dependendo da quantidade de itens.
 - a) Caso o cliente tenha mais que 20 itens, ele entra na menor fila dos caixasG.
 - b) Caso o cliente tenha até 20 itens, ele entra na fila dos caixasP.
3. Fazer um loop que representa o **tempo em um dia**, contando de 0 até 24 via um contador que aumenta de 0.1 em 0.1 unidades de tempo;
 - a) A cada iteração desse loop, o tempo de atendimento dos clientes “na vez” deve ser decrementado em 0.1 até zerar. Uma vez que chegue em 0, a fila correspondente “anda” e se contabiliza mais uma pessoa atendida naquele caixa.
4. Ao fim do loop, calcule a proporção entre a **quantidade de clientes atendidos** e a **quantidade total de clientes** que chegou em cada tipo de fila para saber qual dos dois modelos atendeu uma porcentagem maior de pessoas: caixas com fila coletiva ou com fila individual. Se algum caixa não tiver recebido clientes, considere a porcentagem 100%.

Obs 1. Seu código deve implementar os caixas e clientes da seguinte forma:

```
typedef struct{
    float TAtend; //tempo de atendimento do cliente
    int Qitens; //qtd dos produtos
}Cliente;

typedef struct{
    Cliente fila[20000]; //pessoas na fila do caixa
    int PessoasEsperando; //pessoas que ainda não foram atendidas
    int PessoasAtendidas; //pessoas que já foram atendidas
}CaixaG;

typedef struct{
    int PessoasAtendidas;
    int ocupado; //assume valores 0 ou 1 dependendo se o caixa está ocupado
    Cliente atual; //cliente sendo atendido
}CaixaP;
```

Obs 2. O código a seguir gera números aleatórios como foi pedido:

```
#include <time.h>
#include <stdlib.h>
...
srand( (unsigned)time(NULL) ); // Inicializa o gerador de números aleatórios
float num_sorteado = (rand()%101)/100.0; //sorteia um número entre 0 e 1
int num_sorteado2 = rand()%40 + 1; //sorteia um inteiro entre 1 e 40
```

Não esqueça de comentar o código, para que este fique o mais legível possível! **Boa sorte!!**