

Disciplina	Turma	Nome da Tarefa	Data de Entrega
Linguagem de Programação	ADS-VA2	Esta atividade é um simulado. Vale o esforço, vale o aprendizado. Não vale nota.	Ver prazo na Tarefa do Teams

**Instruções:**

1. **Recomendo que leiam** todas estas instruções **antes** de começar.
2. **Atividade em duplas:** cada dupla faz o seu exercício e apenas um integrante faz a entrega  
As duplas não podem conversar entre si.  
Se alguém não conseguir formar uma dupla, então deverá fazer sozinho.  
Não serão permitidos grupos de três ou mais alunos.
3. **Atividade com consulta:** aos alunos é permitida a consulta a quaisquer materiais disponíveis física ou digitalmente, com exceção a materiais produzidos por colegas de turma.
4. Esta atividade tem 3 questões propostas e você deve fazer os programas em arquivos separados. Na entrega faça o upload dos três arquivos na tarefa do Teams.
5. **Obrigatoriamente** o programa deve **exibir** no topo da tela o **nome completo dos alunos e o número da questão**.  
Sugiro colocar as duas linhas abaixo como primeiros comandos do programa:

```
printf("Zezinho Pereira da Silva\n");  
printf("Huguinho Pereira da Silva\n");  
printf("Questao 999\n");
```

Obviamente os nomes fictícios Zezinho/Huguinho e o 999 nas linhas acima devem ser substituídos, respectivamente, pelos nomes da dupla e pelo número da questão que você está resolvendo (acreditem, em semestres passados eu já recebi programas em que essa troca não foi feita)

---

Disciplina	Turma	Nome da Tarefa	Data de Entrega
Linguagem de Programação	ADS-VA2	Esta atividade é um simulado. Vale o esforço, vale o aprendizado. Não vale nota.	Ver prazo na Tarefa do Teams

**Questão 1:**

Nome do arquivo a ser enviado: q01.cpp

**A primeira coisa que o programa deve fazer são os printf com o nome completo dos alunos e o número da questão**

Vamos criar um jogo de adivinhação de números. São dois jogadores – o computador e o humano – e a ideia é a seguinte: dado um intervalo fechado [Min, Max] o jogador computador sorteia (randomiza) um número que esteja dentro no intervalo. O segundo jogador tem como desafio descobrir qual número foi escolhido. Para isso ele deve escolher um valor e digitá-lo. O computador deve então declarar se está certo ou errado. Se estiver certo, terminou o jogo. Se estiver errado o computador deve informar se o número a ser adivinhado é menor ou maior que o palpite dado e o jogo continua. Ao final é preciso verificar quantos palpites foram dados até que o valor tenha sido adivinhado também quais foram esses palpites. Use um vetor dinâmico para armazenar os palpites.

Requisito a ser observado

Os valores Min e Max são fornecidos pelo usuário e podem ser quaisquer valores desde que Max seja maior que Min + 100 ( $Max > Min + 100$ ). É obrigatório que o programa verifique isso e não deixe jogar se essa condição não for satisfeita

Escreva um programa para implementar esse jogo.

**Caso de Teste**

Sejam os dados iniciais

Min: 1    Max: 500    Número sorteado: 391

Começa o jogo:

Palpite 1: 250

errado: seu palpite deve ser maior

Palpite 2: 375

errado: seu palpite deve ser maior

Palpite 3: 440

errado: seu palpite deve ser menor

Palpite 4: 400

errado: seu palpite deve ser menor

Palpite 5: 385

errado: seu palpite deve ser maior

Palpite 6: 395

errado: seu palpite deve ser menor

Palpite 7: 391

Acertou!!!

Resultado:

foram 7 palpites até você acertar

e os seus palpites foram esses: 250, 375, 440, 400, 385, 395, 391

Disciplina	Turma	Nome da Tarefa	Data de Entrega
Linguagem de Programação	ADS-VA2	Esta atividade é um simulado. Vale o esforço, vale o aprendizado. Não vale nota.	Ver prazo na Tarefa do Teams

## Questão 2

Nome do arquivo a ser enviado: q02.cpp

**A primeira coisa que o programa deve fazer são os printf com o nome completo dos alunos e o número da questão**

Escreva um programa que leia um número inteiro N, obrigatoriamente par e maior ou igual a 6 e menor ou igual a 32.

Se o N for ímpar ou fora da faixa [6, 32] deve ser mostrada a mensagem “O número TAL é inválido”, onde TAL deve ser substituído pelo valor digitado. Neste caso outro valor deve ser lido. Enquanto o usuário do programa insistir em digitar valores inválidos para N, o programa deve permanecer lendo.

Se N for válido o programa deve exibir na tela um desenho formado por asteriscos e espaços em branco, cuja largura total seja exatamente N caracteres, conforme exemplos abaixo. O lado esquerdo do desenho deve estar encostado no lado esquerdo da tela. A espessura da parede deve ser de 2 asteriscos e o espaço interno montado com espaços em branco.

Entrada: N = 19

Saída: O número 19 é inválido

Entrada: N = 6

Saída

```

****
*****
**  **
**  **
*****
****

```

Entrada: N = 10

Saída

```

*****
*****
**      **
**      **
**      **
**      **
**      **
**      **
*****
*****

```

Entrada: N = 12

Saída

```

*****
*****
**      **
**      **
**      **
**      **
**      **
**      **
**      **
*****
*****

```

É obrigatório executar o desenho utilizando-se um algoritmo com laço.

Desenhos fixos implementados diretamente com o comando printf como mostrado abaixo não serão aceitos.

```

if (N == 6) {
    printf(" **** \n");
    printf("*****\n");
    printf("**  **\n");
    printf("**  **\n");
    printf("*****\n");
    printf(" **** \n");
}

```

**Isso funciona, mas não pode.**

Disciplina	Turma	Nome da Tarefa	Data de Entrega
Linguagem de Programação	ADS-VA2	Esta atividade é um simulado. Vale o esforço, vale o aprendizado. Não vale nota.	Ver prazo na Tarefa do Teams

### Questão 3

Nome do arquivo a ser enviado: q03.cpp

**A primeira coisa que o programa deve fazer são os printf com o nome completo dos alunos e o número da questão**

Considere a tabela de produtos e seus dados ao lado. Essa tabela contém códigos de produtos e seus preços unitários

de venda, que podem ocorrer no Varejo ou no Atacado, a depender da quantidade vendida. Assim, as vendas são classificadas em dois grupos: o Grupo Varejo que contém todas as vendas em que a quantidade não ultrapassa a quantidade mínima para atacado (QMA) e o Grupo Atacado que contém as vendas cuja quantidade é maior ou igual à QMA.

Código	Preço Unitário Varejo	Preço Unitário Atacado	Quantidade mínima para atacado (QMA)
16	14,35	12,93	50
23	35,12	29,85	100
27	19,35	16,76	70
34	63,40	58,25	40

Você deve escrever um programa que tenha como entrada inicial um número inteiro que representa o número de vendas realizadas NV, obrigatoriamente maior que zero. Em seguida deverão ser lidos do teclado NV pares de dados de entrada que consistem em código (Cod) e quantidade da venda (QV). Se o código não estiver na tabela deve-se emitir a mensagem “Código inválido”. Se o código for válido, o programa deve calcular o valor de venda definido como Preço Unitário \* QV, no qual deve ser usado o Preço Unitário adequado, conforme a quantidade vendida, QV. Se a QV for menor que a quantidade mínima para atacado (QMA) deve-se usar o preço de varejo, caso contrário deve-se usar o preço de atacado. Cada venda deve ser totalizada no grupo correto, Varejo ou Atacado.

Ao final o programa deve apresentar na tela, separadamente, os totais de vendas para os Grupos Varejo e Atacado, e o valor total de vendas.

Caso de teste

Entrada para NV = 6			Saída	
	Cod	QV	Total de Vendas do Grupo Varejo: R\$ 778,15	
venda 1	27	3	Total de Vendas do Grupo Atacado: R\$ 6033,80	
venda 2	16	60	Vendas Totais: R\$ 6811,95	
venda 3	16	6		
venda 4	23	120		
venda 5	27	100		
venda 6	34	10		