

Questão **1**

Ainda não respondida

Vale 1,00 ponto(s).

Uma nova copiadora irá abrir na sua cidade. Você foi contratado para fazer o programa que gerencia as senhas de utilização dos serviços. As instruções passadas a você são: existem dois tipos de senha, as prioritárias e as normais; as senhas são chamadas de modo a serem chamadas 3 prioritárias para então uma normal. Não havendo prioridades, as pessoas com senha normal podem ser atendidas normalmente. Sempre que um atendimento normal for realizado, os próximos três devem ser prioritários, se possível. Se estiver na hora de realizar um atendimento de cliente normal mas não houver nenhum cliente normal, deve ser chamado um nome do atendimento preferencial, se houver.

Este é um problema de **uso** de fila. Você deverá usar uma estrutura de dados do tipo **fila**, sendo que a classe fila não deve conter nenhum método que tenha características do problema. Em outras palavras, a classe fila não pode ter mecanismos indevidos de acessos a seus dados. Também não é permitido o uso da STL ou similares.

O programa deve ler comandos representados por palavras e seus respectivos argumentos assim:

- O comando "normal" seguido de um nome (sem espaços) representa retirada de uma senha normal.
- O comando "prioridade" seguido de um nome (sem espaços) representa retirada de uma senha prioritária.
- O comando "atender" representa uma senha sendo chamada.
- O comando "fim" solicita o fim da execução do programa.

Para cada comando de atendimento, o programa deverá escrever o nome da pessoa que deve ser atendida, de acordo com as regras de atendimento vistas. Caso não haja cliente a ser chamado, o programa deve escrever "AGUARDE" (letras maiúsculas). A cada atendimento normal, a contagem para prioritários deve ser reiniciada.

Você pode usar o código base disponível no final do enunciado

Entradas:

1. Sequência de comandos, seguindo as instruções acima.

Saídas:

1. Somente o comando "atender" gera saída, conforme instruções acima.

Exemplo de Entrada:

normal Abner
prioridade Ackson
normal Adilson
normal Adriana
normal Adriano
normal Adriel
prioridade Adrielle
atender
atender
atender
atender
prioridade Adriene
prioridade Adson
prioridade Afonso
prioridade Agatha
atender
atender
atender
atender
atender
fim

Exemplo de Saída:

Ackson
Adrielle
Abner
Adilson
Adriene
Adson
Afonso
Adriana
Agatha

Exemplo de Entrada:

prioridade aline
prioridade alice

atender
atender
atender
fim

Exemplo de Saída:

aline
alice
AGUARDE

```
#include <iostream>
using namespace std;
struct dado{
struct dado{
    string nome = "vazio";
    string tipo;
    dado *proximo;
};
class fila{
class fila{
    private:
        dado *inicio;
        dado *fim;
    public:
        fila(){
            inicio = NULL;
            fim = NULL;
        }
        void enfileirar (dado entrada){
            dado *novo = new dado;
            *novo = entrada;
            if (fim != NULL)
                fim->proximo = novo;
            else
                inicio = novo;
            fim = novo;
        }
        dado desenfileirar(){
            if (inicio == NULL) throw runtime_error("Fila vazia!");
            dado *retornado = new dado;
            dado *removido = inicio;
            if (inicio == fim)
                fim = NULL;
            retornado->nome = inicio->nome;
            retornado->tipo = inicio->tipo;
            inicio = inicio->proximo;
            removido->proximo = NULL;
            delete removido;
            return *retornado;
        }
}
```

```
~fila(){
    while (inicio != NULL)
        desenfileirar();
}
dado atendimento (string tipo){
// FAÇA SEU CÓDIGO AQUI
}
};

int main(){
    fila banco;
    string comando, tipo;
    dado entrada, atendido;
    cin>>comando;
    while (comando != "fim"){
        if (comando == "normal"){
            cin>>entrada.nome;
            entrada.tipo = "normal";
            banco.enfileirar(entrada);
        } else if (comando == "prioridade"){
            cin>>entrada.nome;
            entrada.tipo = "prioridade";
            banco.enfileirar(entrada);
        } else if (comando == "atender"){
            //COMPLETE AQUI
        }
        cin>>comando;
    }
    return 0;
}
```

Tamanho máximo para arquivos: 1 Gb, número máximo de anexos: 1



[Arquivos](#)



Você pode arrastar e soltar arquivos aqui para adicioná-los.

Questão **2**

Ainda não respondida

Vale 1,00 ponto(s).

Considere os TADs e métodos estudados para a implementação de uma pilha. Implemente um método *inverter()* que, dada a chamada *P1.inverter()*, a ordem dos elementos de *P1* é invertida. **Os elementos da pilha devem continuar armazenados em *P1* após a inversão.**

Você pode utilizar o código base fornecido no enunciado.

Dica: utilize estruturas auxiliares para a inversão.

Entradas:

1. Número de elementos a serem inseridos em *P1*.
2. Elementos de *P1* em ordem de inserção

Saídas:

1. Elementos da pilha *P1*, após inversão

Exemplo de Entrada 1:

```
5
3 6 8 1 9 (elementos impressos em ordem de inserção)
```

Exemplo de Saída1:

```
3 6 8 1 9 (elementos desempilhados após a inversão da pilha P1)
```

Exemplo de Entrada 2:

```
10
5 1 2 3 8 6 1 9 12 19 (elementos impressos em ordem de inserção)
```

Exemplo de Saída2:

```
5 1 2 3 8 6 1 9 12 19 (elementos desempilhados após a inversão da pilha P1)
```

Exemplo de Entrada 3:

23

1 3 5 7 9 0 2 4 6 8 11 13 15 17 19 10 12 14 16 18 21 23 25 (elementos impressos em ordem de inserção)

Exemplo de Saída3:

1 3 5 7 9 0 2 4 6 8 11 13 15 17 19 10 12 14 16 18 21 23 25 (elementos desempilhados após a inversão da pilha P1)
--

```
class pilha{
    private:
        dado *topo;
    public:
        pilha(){
            topo = NULL;
        }
        void empilha (dado entrada){
            dado *novo = new dado;
            *novo = entrada;
            novo->proximo = topo;
            topo = novo;
        }
        int desempilha(){
            if (topo == NULL) throw runtime_error("Pilha vazia!");
            int removido;
            dado *apagado = topo;
            removido = topo->elemento;
            topo = topo->proximo;
            apagado->proximo = NULL;
            delete apagado;
            return removido;
        }
        ~pilha(){
            while (topo != NULL)
                desempilha();
        }
        void inverte(){
            // FAÇA SEU CÓDIGO PARA INVERSÃO DA PILHA
        }
        void imprime(){
            // FAÇA SEU CÓDIGO PARA IMPRESSÃO DA PILHA
        }
};

int main(){
    int tamanho;
    cin>> tamanho;
    dado entrada;
    pilha mp;
    for (int i=0; i<tamanho; i++){
        cin>>entrada.elemento;
```

```
        mp.empilha(entrada);  
    }  
    mp.inverte();  
    mp.imprime();  
    return 0;  
}
```

Tamanho máximo para arquivos: 1 Gb, número máximo de anexos: 1



[Arquivos](#)



Você pode arrastar e soltar arquivos aqui para adicioná-los.

Questão **3**

Ainda não respondida

Vale 1,00 ponto(s).

Você tem uma estrutura Heap com dados de operações e, cada operação tem sua prioridade. Os dados são organizados em uma MaxHeap de acordo com a prioridade. Implemente um método que altere a prioridade de um dado elemento e reorganize a estrutura de modo que os dados com a nova prioridade fiquem corretos.

```
#include <iostream>
#include <cstdlib>
#include <cstring>
using namespace std;

struct dado{
    char operacao;
    int prioridade;
};

class MaxHeap {
private:
    dado* heap;
    int capacidade;
    int tamanho;
    inline int pai(int i);
    inline int esquerdo(int i);
    inline int direito(int i);
    void arruma(); // heapify()
    void corrigeDescendo(int i);
    void corrigeSubindo(int i);
public:
    MaxHeap(int cap);
    ~MaxHeap();
    void imprime();
    dado retiraRaiz();
    void insere(dado d);
};
```

```
MaxHeap::MaxHeap(int cap) {
    capacidade = cap;
    heap = new dado[cap];
    tamanho = 0;
}
MaxHeap::~MaxHeap() {
    delete[] heap;
}
void MaxHeap::arruma() {
    for (int i=(tamanho/2-1); i>=0; i--) {
        corrigeDescendo(i);
    }
}
int MaxHeap::pai(int i) {
    return (i-1)/2;
}

int MaxHeap::esquerdo(int i) {
    return 2*i+1;
}

int MaxHeap::direito(int i) {
    return 2*i+2;
}

void MaxHeap::corrigeDescendo(int i) {
    int esq = esquerdo(i);
    int dir = direito(i);
    int maior = i;

    if ((esq < tamanho) and (heap[esq].prioridade > heap[maior].prioridade)) {
        maior = esq;
    }

    if ((dir < tamanho) and (heap[dir].prioridade > heap[maior].prioridade)) {
        maior = dir;
    }

    if (maior!=i) {
        swap(heap[i], heap[maior]);
        corrigeDescendo(maior);
    }
}
```

```
    }  
}  
void MaxHeap::corrigeSubindo(int i) {  
    int p = pai(i);  
    if (heap[i].prioridade > heap[p].prioridade) {  
        swap(heap[i], heap[p]);  
        corrigeSubindo(p);  
    }  
}  
  
void MaxHeap::imprime() {  
    for (int i=0; i<capacidade; i++) {  
        cout << heap[i].operacao << " " << heap[i].prioridade<<endl;  
    }  
    cout << endl;  
}  
  
dado MaxHeap::retiraRaiz() {  
    if (tamanho == 0){  
        cerr << "Erro ao retirar raiz" << endl;  
        exit(EXIT_FAILURE);  
    }  
    dado aux = heap[0];  
    swap(heap[0], heap[tamanho-1]);  
    tamanho--;  
    corrigeDescendo(0);  
    return aux;  
}  
  
void MaxHeap::insere(dado d){  
    if (tamanho == capacidade) {  
        cerr << "Erro ao inserir" << endl;  
        exit(EXIT_FAILURE);  
    }  
    heap[tamanho] = d;  
    corrigeSubindo(tamanho);  
    tamanho++;  
}  
  
int main () {  
    int capacidade, opcao;  
    cin>> capacidade;  
  
    MaxHeap *h = new MaxHeap(capacidade);
```

```
    dado entrada,saida;
    cin >> opcao;
    while (opcao != 0){
        if (opcao == 1){
            cin>>entrada.operacao>>entrada.prioridade;
            h->insere(entrada);
        } else if (opcao == 2){
            saida = h->retiraRaiz();
            cout<<saida.operacao<<" "<<saida.prioridade<<endl;
        } else if (opcao == 3){//APENAS PARA EFEITO DE DEPURAÇÃO
            h->imprime();
        } else if (opcao == 4){
            //COMPLETE O CÓDIGO
        }
        cin >> opcao;
    }

    delete h;

    return 0;
}
```

Tamanho máximo para arquivos: 1 Gb, número máximo de anexos: 1



[Arquivos](#)



Você pode arrastar e soltar arquivos aqui para adicioná-los.

Questão 4

Ainda não respondida

Vale 1,00 ponto(s).

Considerando que a classe Fila já está implementada, sua tarefa é implementar o destrutor **iterativo** para a Fila. Qual das alternativas contém o trecho correto de código?

Escolha uma opção:

☐ a.

```
while ( not this->inicio == NULL ) {  
    noh* umNoh = this->inicio;  
    this->inicio = this->inicio->proximo;  
    delete umNoh;  
    this->tamanho--;  
}
```

☐ b.

```
while ( this->inicio != NULL ) {  
    noh* umNoh = this->inicio;  
    delete umNoh;  
    this->tamanho--;  
    this->inicio = this->inicio->proximo;  
}  
  
this->fim = NULL;
```

☐ c.

```
while ( not this->inicio == NULL ) {  
    noh* umNoh = this->inicio;  
    umNoh->proximo = this->inicio->proximo;  
    delete umNoh;  
    this->tamanho--;  
}
```

☐ d.

```
while ( not this->inicio == NULL ) {  
    noh* umNoh = this->inicio;  
    this->inicio = this->inicio->proximo;  
    delete umNoh;  
    this->tamanho--;  
}  
this->fim = NULL;
```

☐ e.

```
if ( this->inicio != NULL ) {  
    Noh* umNoh = this->inicio;  
    this->inicio = this->inicio->proximo;  
    delete umNoh;  
    this->tamanho--;  
    destrutor_rec();  
} else  
    this->fim = NULL;
```

Questão **5**

Ainda não respondida

Vale 1,00 ponto(s).

Em uma operação de remoção em um Heap, qual elemento é sempre removido?

Escolha uma opção:

- ☐ a. Qualquer elemento pode ser escolhido para remoção
- ☐ b. O menor elemento
- ☐ c. O elemento na raiz
- ☐ d. O elemento do último Noh

Questão 6

Ainda não respondida

Vale 1,00 ponto(s).

Considere as pilhas A e B, sendo $A = [1, 3, 5, 6]$ e $B = [7, 9, 10]$, com o elemento mais a direita sendo o último a ser empilhado. Quais as configurações de A e B após a execução do conjunto de passos abaixo?

- Desempilhar de A;
- Empilhar, em B, o elemento desempilhado de A;
- Empilhar, em B, o elemento 8;
- Desempilhar de B;
- Empilhar, em A, o elemento desempilhado de B;
- Empilhar, em B, o elemento 2;
- Desempilhar de A;
- Empilhar, em B, o elemento desempilhado de A.

Escolha uma opção:

- ☐ a. $A = [1, 3, 5]$ e $B = [7, 9, 10, 6, 2, 8]$
- ☐ b. $A = [1, 3, 5, 8]$ e $B = [7, 9, 10, 6, 2]$
- ☐ c. $A = [1, 3, 5, 8]$ e $B = [2, 6, 7, 9, 10]$
- ☐ d. $A = [1, 3, 5]$ e $B = [8, 2, 6, 7, 9, 10]$

Questão 7

Ainda não respondida

Vale 1,00 ponto(s).

Sejam $A = [1, 2, 3, 4]$, $B = [5, 6, 7]$ e $C = [8, 9, 10]$ três filas, em que o início encontra-se à esquerda e o final à direita. Suponha agora que sejam realizadas **duas vezes** o seguinte conjunto de operações:

- > Desenfileirar em A
- > Enfileirar em B o elemento desenfileirado de A
- > Enfileirar em C o elemento desenfileirado de A
- > Desenfileirar em B
- > Desenfileirar em C
- > Enfileirar em A o elemento desenfileirado de C
- > Enfileirar em B o elemento desenfileirado de C

Qual opção apresenta as filas resultantes?

Escolha uma opção:

- ☐ a. $A = [2, 1, 10, 9]$
 $B = [5, 4, 10, 3, 9]$
 $C = [8, 4, 3]$
- ☐ b. $A = [9, 9, 2, 2]$
 $B = [2, 9, 9, 4, 4, 2, 2]$
 $C = [2, 4, 4]$
- ☐ c. $A = [2, 3, 4, 8]$
 $B = [6, 7, 1, 8]$
 $C = [9, 10, 1]$
- ☐ d. $A = [3, 4, 9, 9]$
 $B = [7, 2, 2, 9, 9]$
 $C = [10, 2, 2]$

- ☐ e. $A = [3, 4, 8, 9]$
 $B = [7, 1, 8, 2, 9]$
 $C = [10, 1, 2]$

Questão 8

Ainda não respondida

Vale 1,00 ponto(s).

Considere o seguinte **maxheap**

35	32	17	21	11	16	7	9	8	10	3	4
----	----	----	----	----	----	---	---	---	----	---	---

Realize as seguintes operações:

insere o valor 6

remove a raiz;

insere o valor 12;

insere o valor 18;

remove a raiz;

Qual será a configuração do arranjo após a execução das operações indicadas?

Escolha uma opção:

- ☐ a.

32	21	18	9	11	16	17	6	8	10	3	4	12	7
----	----	----	---	----	----	----	---	---	----	---	---	----	---
- ☐ b.

21	11	18	9	10	16	17	6	8	7	3	4	12
----	----	----	---	----	----	----	---	---	---	---	---	----
- ☐ c.

21	11	18	16	10	12	17	6	8	7	3	4	9
----	----	----	----	----	----	----	---	---	---	---	---	---
- ☐ d.

21	11	18	12	10	16	17	6	8	7	3	4	9
----	----	----	----	----	----	----	---	---	---	---	---	---

☐ e.

32	21	17	9	11	16	7	6	8	10	3	4
----	----	----	---	----	----	---	---	---	----	---	---

Questão 9

Ainda não respondida

Vale 1,00 ponto(s).

Considere a seguinte pilha A = [0, 2, 11, 7, 8, 5], sendo que o elemento mais a direita foi o último a ser inserido na pilha, e uma pilha B, inicialmente vazia. Quais seriam as operações necessárias para que a pilha A fique com a seguinte configuração A = [11, 8, 2, 5]?

Escolha uma opção:

- ☐ a.
- Desempilhar elemento em A;
 - Desempilhar elemento em A;
 - Empilhar em B o elemento que acabou de ser desempilhado em A;
 - Desempilhar elemento em A;
 - Empilhar em B o elemento que acabou de ser desempilhado em A;
 - Desempilhar elemento em A;
 - Empilhar em B o elemento que acabou de ser desempilhado em A;
 - Desempilhar elemento em A;
 - Desempilhar elemento em A;
 - Desempilhar elemento em B;
 - Empilhar em A o elemento que acabou de ser desempilhado em B;
 - Desempilhar elemento em B;
 - Empilhar em A o elemento que acabou de ser desempilhado em B;
 - Empilhar em A o elemento de valor 2;
 - Desempilhar elemento em B;
 - Empilhar em A o elemento que acabou de ser desempilhado em B;
- ☐ b.
- Desempilhar elemento em A;
 - Empilhar em B o elemento que acabou de ser desempilhado em A;
 - Desempilhar elemento em A;
 - Empilhar em B o elemento que acabou de ser desempilhado em A;
 - Desempilhar elemento em A;
 - Empilhar em B o elemento que acabou de ser desempilhado em A;

- Desempilhar elemento em A;
 - Empilhar em B o elemento que acabou de ser desempilhado em A;
 - Desempilhar elemento em A;
 - Desempilhar elemento em A;
 - Desempilhar elemento em B;
 - Empilhar em A o elemento que acabou de ser desempilhado em B;
 - Desempilhar elemento em B;
 - Desempilhar elemento em B;
 - Empilhar em A o elemento que acabou de ser desempilhado em B;
 - Empilhar em A o elemento de valor 2;
 - Desempilhar elemento em B;
 - Empilhar em A o elemento que acabou de ser desempilhado em B;
- ☐ c.
- Desempilhar elemento em A;
 - Empilhar em B o elemento que acabou de ser desempilhado em A;
 - Empilhar em B o elemento de valor 2;
 - Desempilhar elemento em A;
 - Empilhar em B o elemento que acabou de ser desempilhado em A;
 - Desempilhar elemento em A;
 - Empilhar em B o elemento que acabou de ser desempilhado em A;
 - Desempilhar elemento em A;
 - Empilhar em B o elemento que acabou de ser desempilhado em A;
 - Desempilhar elemento em B;
 - Empilhar em A o elemento que acabou de ser desempilhado em B;
 - Desempilhar elemento em B;
 - Desempilhar elemento em B;
 - Empilhar em A o elemento que acabou de ser desempilhado em B;
 - Desempilhar elemento em B;
 - Empilhar em A o elemento que acabou de ser desempilhado em B;
- ☐ d.
- Desempilhar elemento em A;
 - Empilhar em B o elemento que acabou de ser desempilhado em A;
 - Desempilhar elemento em A;
 - Empilhar em B o elemento que acabou de ser desempilhado em A;
 - Desempilhar elemento em A;
 - Empilhar em B o elemento que acabou de ser desempilhado em A;

- Desempilhar elemento em A;
- Empilhar em B o elemento que acabou de ser desempilhado em A;
- Desempilhar elemento em A;
- Desempilhar elemento em B;
- Empilhar em A o elemento que acabou de ser desempilhado em B;
- Desempilhar elemento em B;
- Empilhar em A o elemento que acabou de ser desempilhado em B;
- Empilhar em A o elemento de valor 2;
- Desempilhar elemento em B;
- Empilhar em A o elemento que acabou de ser desempilhado em B;

Questão **10**

Ainda não respondida

Vale 1,00 ponto(s).

Considere a seguinte fila A = [1, 2, 3, 4, 5, 6], sendo que o elemento de valor 1 foi o primeiro a ser inserido na fila. Quais seriam as operações necessárias para que a fila A fique com a seguinte configuração A = [5, 6, 2, 3, 7, 4]?

Escolha uma opção:

- ☐ a.
- Desenfileirar elemento;
 - Desenfileirar elemento;
 - Enfileirar elemento que acabou de ser desenfileirado;
 - Desenfileirar elemento;
 - Desenfileirar elemento;
 - Enfileirar elemento que acabou de ser desenfileirado;
 - Desenfileirar elemento;
 - Desenfileirar elemento;
 - Enfileirar elemento que acabou de ser desenfileirado;
 - Enfileirar novo elemento;
 - Enfileirar novo elemento;
- ☐ b.
- Desenfileirar elemento;
 - Desenfileirar elemento;
 - Enfileirar elemento que acabou de ser desenfileirado;
 - Desenfileirar elemento;
 - Enfileirar elemento que acabou de ser desenfileirado;
 - Enfileirar novo elemento;
 - Desenfileirar elemento;
 - Enfileirar elemento que acabou de ser desenfileirado;
- ☐ c.
- Desenfileirar elemento;
 - Desenfileirar elemento;
 - Desenfileirar elemento;

- Enfileirar elemento que acabou de ser desenfileirado;
 - Enfileirar novo elemento;
 - Enfileirar novo elemento;
 - Desenfileirar elemento;
 - Enfileirar elemento que acabou de ser desenfileirado;
- ☐ d.
- Desenfileirar elemento;
 - Enfileirar elemento que acabou de ser desenfileirado;
 - Enfileirar novo elemento;
 - Enfileirar novo elemento;
 - Desenfileirar elemento;
 - Enfileirar elemento que acabou de ser desenfileirado;
 - Desenfileirar elemento;

Questão **11**

Ainda não respondida

Vale 1,00 ponto(s).

Considerando o pseudo-código seguinte, qual é a funcionalidade do programa?

```
// Considere que a classe Pilha está implementada.  
void funcao_a(Pilha minhaPilha) {  
    while ( not minhaPilha.empty() )  
        cout << minhaPilha.desempilha() << ' ' ;  
}  
  
int main () {  
    Pilha minhaPilha;  
    minhaPilha.empilha(0);  
    minhaPilha.empilha(1);  
    int n = 10, a = 0, b = 0;  
    for (int i = 0; i < n; i++)  
    {  
        a = minhaPilha.desempilha();  
        b = minhaPilha.desempilha();  
        minhaPilha.empilha(b);  
        minhaPilha.empilha(a);  
        minhaPilha.empilha(a + b);  
    }  
  
    funcao_a(minhaPilha);  
}
```

Escolha uma opção:

- ☐ a. Imprimir a soma dos número de 0 até n-1
- ☐ b. Imprimir os n primeiros números na sequência de fibonacci na ordem inversa

- ☐ c. Imprimir uma sequência de números sem sentido aparente
- ☐ d. Imprimir os n primeiros números na sequência de fibonacci

Questão **12**

Ainda não respondida

Vale 1,00 ponto(s).

Considere o seguinte pseudocódigo para construção de um **maxheap** a partir de um vetor:

```
constroiHeap(vetor, tamanho): //heapify()
dados ← copiaDados(vetor);
// INICIO informa posição inicial utilizada no vetor
para todo i de METADE até INICIO {
    corrigeDescendo(i);
}
```

Suponha também o vetor V dado por:

8	4	9	13	6	12	15	1	21	0	30
---	---	---	----	---	----	----	---	----	---	----

Qual será a configuração do vetor de dados do heap após **quatro passos** de execução do comando para todo no pseudocódigo apresentado?

Escolha uma opção:

- ☐ a.
- | | | | | | | | | | | |
|---|----|----|----|---|----|---|---|----|---|---|
| 8 | 30 | 15 | 21 | 6 | 12 | 9 | 1 | 13 | 0 | 4 |
|---|----|----|----|---|----|---|---|----|---|---|
- ☐ b.
- | | | | | | | | | | | |
|---|----|----|----|---|---|----|---|----|---|---|
| 8 | 30 | 12 | 21 | 4 | 9 | 15 | 1 | 13 | 0 | 6 |
|---|----|----|----|---|---|----|---|----|---|---|
- ☐ c.
- | | | | | | | | | | | |
|---|----|----|----|---|----|---|---|----|---|---|
| 8 | 30 | 15 | 21 | 4 | 12 | 9 | 1 | 13 | 0 | 6 |
|---|----|----|----|---|----|---|---|----|---|---|
- ☐ d.
- | | | | | | | | | | | |
|----|----|----|----|---|---|----|---|---|---|---|
| 30 | 21 | 12 | 13 | 6 | 9 | 15 | 1 | 8 | 0 | 4 |
|----|----|----|----|---|---|----|---|---|---|---|

☐ e.

8	30	12	21	6	9	15	1	13	0	4
---	----	----	----	---	---	----	---	----	---	---