**ISHMEET KAUR | 100809450**

**RISHAB SINGH | 100787473**

**OS TUTORIAL #2 - CRN74027**

**Product idea:  
  
  
a glove for athletes that’s monitors how they follow through with their fingers depending on how theyre moving + how that shot goes in (i.e is it a swish? Does it backbackord?/ airball?. )**

* **Could look like thoe. Rubber finger strengthening temu things.**

Conceptual Questions

1. List each of the modes for the fopen function to the perform the following  
operations: read, write, read and write, append to a file.

* R**ead**: **"r"** Opens a file for reading. The file must exist.
* **Write**: **"w"** Creates an empty file for writing. If a file with the same name exists, its content is erased.
* **Read and Write**: **"r+"** Opens a file for both reading and writing. The file must exist.
* **Append**: **"a"** Opens a file for appending. Data written to the file is added to the end. The file is created if it does not exist.

2. Does dynamic memory use the stack or heap? What is the difference between the stack and heap?

1. **Dynamic Memory: Stack vs. Heap**
   * **Dynamic memory** allocation uses the **heap**. The **stack** is used for static memory allocation, including function calls, local variables, and control flow.
   * **Stack**: Memory is managed automatically by the compiler. Memory allocation and deallocation are done in a last-in, first-out manner. It's fast but limited in size.
   * **Heap**: Memory is managed by the programmer (using **malloc**, **calloc**, **free**, etc.). It's more flexible and can accommodate larger amounts of data, but improper use can lead to memory leaks or fragmentation.

3. Explain what a pointer is, and provide examples (in C code) of how to change the address that a pointer points to and how to access the data the pointer points to.

* A pointer is a variable that stores the memory address of another variable. Pointers are used for dynamic memory management, passing functions as arguments, and operating on arrays.

4. Read the documentation on the malloc and free functions and explain  
briefly how to use malloc.

(memory allocation) is used to dynamically allocate a block of memory on the heap. The function returns a pointer to the allocated memory, or NULL if the allocation fails.

5. What is the difference between malloc and calloc?

* **malloc** allocates a single block of memory of a specified size and does not initialize the memory. The contents of the allocated memory are indeterminate.
* **calloc** (contiguous allocation) allocates memory for an array of elements, initializes all bits to zero, and returns a pointer to the allocated memory. Syntax: **calloc(numberOfElements, elementSize)**.

Program1.c:  
  
A screenshot of a computer

Description automatically generated

Program2.c

A screenshot of a computer

Description automatically generated

Program3.cA screen shot of a computer

Description automatically generated

Program4.c

A screenshot of a computer program

Description automatically generated

Program5.c

A screenshot of a computer program

Description automatically generated

File Structure and content:  
A screenshot of a computer

Description automatically generated

**Program Management**

* **Definition**: Coordinated management of a portfolio of projects to achieve benefits that individual project management cannot.
* **Strategic Programs**: Implement a single strategy, like merging IT systems of organizations.
* **Business Cycle Programs**: Projects scheduled within a specific timeframe like a financial year.
* **Infrastructure Programs**: Different applications sharing the same hardware/software infrastructure.
* **Research and Development Programs**: Balance between high-risk, high-reward projects and lower-risk projects for long-term success.
* **Innovative Partnerships**: Collaborative efforts to develop new technologies that benefit a range of companies.

**Programme Managers vs Project Managers**

* **Programme Managers**:
  + Handle multiple projects simultaneously.
  + Maintain personal relationships with skilled resources.
  + Optimize resource use.
  + See projects as similar entities.
* **Project Managers**:
  + Focus on one project at a time.
  + Have impersonal relationships with resources.
  + Aim to minimize resource demands.
  + View projects as unique.

**Strategic Program Planning**

* **Program Mandate**: Describes new services/capabilities and organizational improvements.
* **Program Brief**: Acts like a feasibility study, emphasizing costs and benefits.
* **Vision Statement**: Explains the organization's new capabilities post-program.
* **Blueprint**: Outlines changes needed to achieve new capabilities.

**Benefits Management**

* **Process**:
  + Define expected benefits.
  + Analyze the balance between costs and benefits.
  + Plan for benefits realization.
  + Assign responsibilities for benefits.
  + Monitor benefits achievement.
* **Benefits Types**:
  + Mandatory requirements.
  + Improved quality of service.
  + Increased productivity.
  + Motivated workforce.
  + Internal management benefits.
  + Risk reduction.
  + Economies.
  + Revenue enhancement/acceleration.
  + Strategic alignment.

**Quantifying Benefits**

* **Quantified and Valued**: Direct financial benefits like staff reduction savings.
* **Quantified but not Valued**: Indirect benefits like reduced customer complaints.
* **Identified but not Easily Quantified**: Intangible benefits like public approval.

**Important Considerations**

* **Project Failure**: Can occur if the wrong project is started, not just due to poor management.
* **Profitability vs. Opportunity Cost**: Even profitable projects may not be the best use of resources if there are more lucrative options.
* **Risk and Return**: High potential return projects often carry the most risk.