

## Homework 2

### Has0027

**1.) Advantages:** Chained allocation keeps the pointers in the end of each block, so there is no need for a separate block to be wasted for an index table.

**Disadvantages:** Chained allocation does not allow random access since we would have to sequentially search through each block in a chain for access.

**2.)** An FST would use up as much space as it takes to hold a value for each block. This means that the more blocks we have, the more space we will need to provide the FST to hold these values.

**Solution:** We are currently holding a value for each block that tells us whether it is allocated or unallocated. So instead we can just have the FST have a starting block and how many blocks after that are free. This will allow us to tell how much free space we have without having to use up space to keep track of it.

**3.)** When programs have overlapping memory addresses, programs can be slowed down due to the affected function calls. We can fix this issue by relocating and updating the affected memory addresses.

**4.) A Physical Address** points to a physical amount of allocated memory in the memory itself. A **Logical Address** is a reference point that is used to point to the physical memory location. A **Relative Address** is an address that is specified by calculating its distance from another address.

**5.) First-fit** finds the first memory location that contains enough space required to fit the allocated block starting from the top of the memory. This method is very simple meaning it is very easy to implement. The downside to this method is since it is trying to find the first possible space, the memory is mostly going to be top biased. This means that it will become slower as more memory is being used since most of the free space will be towards the bottom.

**Best-fit** finds the memory location that will leave the least amount of unused space in the allocated block. This method is very efficient as it will always leave very small amounts of unused space for each block allocated. The downside to this is that it is very slow since it has to scan the entire file system to find the perfect fit. It also creates these very small gaps in the memory that are so small that almost nothing can fit in them.

**Next-fit** finds the first memory location that contains enough space required to fit the allocated block starting from the block that was last allocated. This method is much faster than Best-fit and does not have a top bias like First-fit. The downside to this is that it is much more complex to implement as we have to keep track of the last block allocated.

**6.)** In order for this process to be loaded into the memory we would first need to relocate processes that are already loaded in the memory to make room for the new process. Doing this is extremely time consuming because all absolute addresses must be updated.