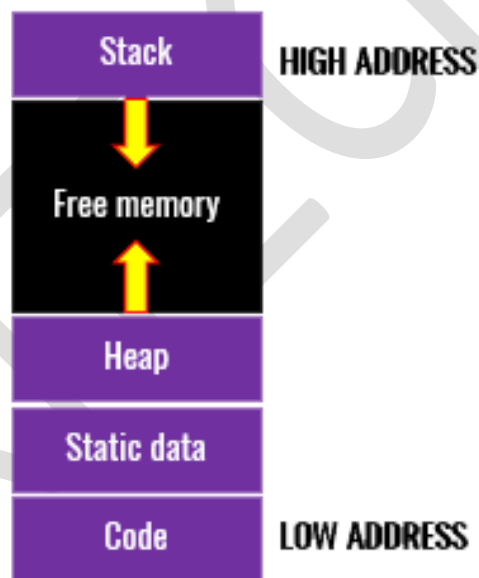


Storage Allocation

- There are basically three storage-allocation strategy is used in each of the three data areas in the organization.
- Static allocation lays out storage for all data objects at compile time.
- Stack allocation manages the run-time storage as a stack,
- Heap allocation allocates and de-allocates storage as needed at run time from a data area known as a heap.

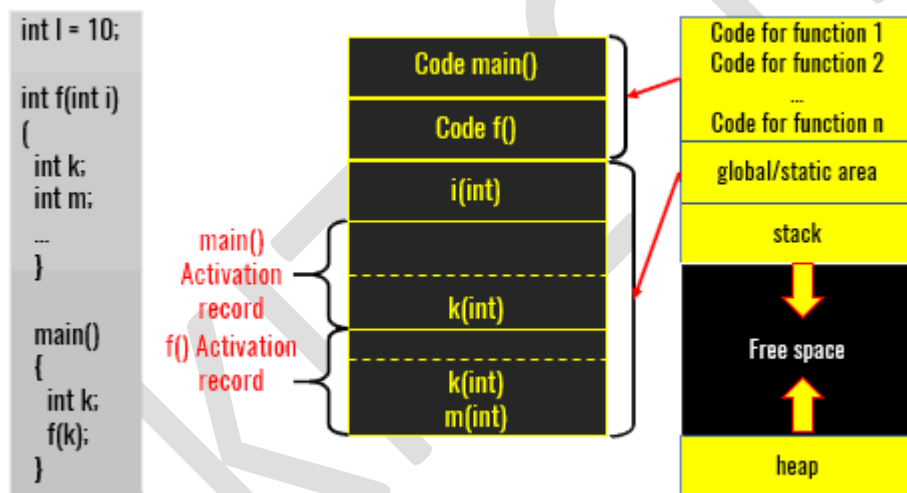


Static Allocation

- In static allocation, names are bound to storage as the program is compiled, so there is no need for a run-time support package.
- Since the bindings do not change at run time, every time a procedure is activated, its names are bound to the same storage locations.
- The above property allows the values of local names to be retained across

activations of a procedure. That is, when control returns to a procedure, the values of the locals are the same as they were when control left the last time.

- From the type of a name, the compiler determines the amount of storage to set aside for that name.
- The address of this storage consists of an offset from an end of the activation record for the procedure.
- The compiler must eventually decide where the activation records go, relative to the target code and to one another.



For More Details Click Here:

<https://www.wikitechy.com/tutorials/compiler-design/storage-allocation>