14 march

① We can not assign a string using assignment operator(=)
for assigning a string we must use ~~stry~~ "strcpy"

② we can not assign a string using assignment operater(=)
for assigning a string we must use "strcpy"

③ we can not assign a string using assignment operator(=)
for assigning a string we must use "strcpy"

④ We can not assign a string using assignment
operator (=) for assigning a string we must use
"strcpy"

⑤ we can not assign a string using assignment
operator (=) for assigning a string we must
use "strcpy"

⑥ We can not assign a string using assignment
operater (=) For assigning a string we must
use ("strcpy")

⑦ We can not assign a string using assignment
opercater(=) for assigning a string we must
use ("strcpy")

(8) we can not assign a string using assignment operator (=) for assigning string. we must use "strcpy"

(9) We can not asign a string using asisnment operator (=) for assigning string. we must use "strcpy"

(10) We can not assign a string using asisnment operater (=) for asisning string we must use "strcpy"

14march

① Malloc receives no. of bytes as input it ~~allow~~ allocates those many number of byte on heap sequentially & returns string/base address in a void pointer

void * malloc (int);

② malloc receives no. of bytes as input it allocates those many number of byte on heap sequentially & returns string/base address in a void pointer.

void * malloc (int);

③ malloc receives no of bytes as input it allocates those many number of byte on heap sequentially & returns string/base address in a void pointer

void* malloc (int);

④ malloc receives no of bytes as input it allocates those many number of byte on heap sequentially & returns string/base address in a void pointer.

void* malloc (int).

⑤ malloc receives no. of bytes as input it allocates those many number of byte on heap sequentially & returns string/base address in a void pointer

void * malloc (int).

⑥ malloc receives no. of bytes as input it allocates those many number of bytes on heap sequentially & returns string/base address in a void pointer.

void * malloc (int);

⑦ malloc receives no of bytes as input it allocates those many number of byte on heap sequentially & return starting / base address in a void pointers

Void* malloc (int);

⑧ malloc receives no. of bytes as input it allocates those many number of byte on heap sequentially & return starting / base address in a void pointers.

Void* malloc(int);

⑨ malloc receives no. of bytes as input it allocates those many number of byte on heap sequentially & return starting / base address in a void pointers

Void* malloc (int);

⑩ malloc receives no. of bytes as input it allocates those many number of byte on heap sequentially & return starting / base address in a void pointers.

Void* malloc (int);