1. Cocher les masques d'adresses IPV4 valides

255.0.0.0 256.0.0.0 192.0.0.0 100.100.100.0 128.0.0.0 192.168.0.0

2. Ecrire la longueur de l'adresse réseau d'une adresse IPv4 ayant le network mask : 255.255.128.0

3. citer les couches suivant le standard TCP/IP

3. dans quelle couche se toruve le protocole HTTP

5. Cocher le(s) protocole(s) de la couche transport suivant le standard TCP/IP

6. Qu'elle est le caractère qui permet de référencer une commande déjà passée dans un termina shell

7. Déterminer la sortie du programme suivant:

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main() {

    char theStr[18] = "SST@sagemcom.com";

    theStr[10]='\10';

    theStr[11]= '\13';

    theStr[12]= '\0';

    theStr[13]= '\7';

    printf("%d",strlen(theStr));

}

8. Déterminer la sortie du programme suivant:

#include <stdio.h>

int main() {

const float PI = 3.14;

float \*q = &PI;

\*q = PI\*10;

printf("%.2f\n",PI);

}

9. Déterminer la ligne en erreur (ecrire 0 si pas d'erreur) :

1. int callFunct(int \*i){

2. (\*i)+=10;

3. return 10;

4. }

5.

6. int main() {

7. int j = 14;

8. callFunct(j);

9. printf( "%d\n", j);

10 }

10. Déterminer la ligne en erreur qui cause le plantage :

1. #include <stdio.h>

2. #include <string.h>

3.

4. int main() {

5. char \*thestar = "sun.milkyway.com";

6. char themoon[] = "ganymed.jupiter.com";

7. printf( "%s-%s\n", thestar, themoon);

8. strcpy(thestar,"sun@milkyway.com");

9. printf( "%s", thestar);

10. themoon[7] = '@';

11. printf("%s", themoon);

12. }

11. Déterminer la sortie du programme suivant:

#include <stdio.h>

int main() {

int k=18;

int &pk = i;

pk+=10;

printf("%d\n", \*pk);

}

12. Déterminer la sortie du programme suivant:

#include <stdio.h>

int main() {

int i=104;

i++;

int \*p = malloc(sizeof(int));

\*p = i;

printf("=>%d\n",p);

free(p);

}

13. Donner exactement le résultat du code C suivant :

int main() {

    int num = 1500;

    if (num >= 0)

        if (num >= 1600)

            printf("a");

        else if (num >= 1500)

            printf("b");

        else if (num >= 1000)

            printf("\*c");

    if (num >= 100)

        printf("d");

    if (num >= 10)

        printf("e");

    else

        printf("f");

    printf("g");

    return 0;

}

14. Donner l'affichage de i dans le programme C suivant :

#include <stdio.h>

int main() {

    printf("Hello world!\n");

    int num = 1500;

    if (num >= 0)

        if (num >= 1600)

            printf("\*");

        else if (num >= 1500)

            printf("\*");

        else if (num >= 1000)

            printf("\*");

        else if (num >= 100)

            printf("\*");

        else if (num >= 10)

            printf("\*");

        else

            printf("#");

    else

        printf("!");

}

15. Donner l'affichage du programme C suivant

void change(char &c) {

c = 'X';

}

int main() {

char c = 'R';

change(&c);

printf("%c\n",c);

return 0;

}

16. Donner l'affichage du programme C suivant

void change(char c) {

c = 'X';

}

int main() {

char c = 'R';

change(c);

printf("%c\n",c);

return 0;

}

17. Donner l'affichage du programme C suivant

void change(char \*c) {

\*c = 'X';

}

int main() {

char c = 'R';

change(&c);

printf("%c\n",c);

return 0;

}

18. Donner le nombre de fois l'affichage de "Greater"

#include <stdio.h>

#include <stdlib.h>

void compute(int val){

extern int \_cnt;

\_cnt += 2;

if (\_cnt > 4)

printf("Greater!\n");

}

int main() {

compute(1);

compute(2);

compute(3);

return 0;

}

int \_cnt = 2;

19. Donner le nombre de fois l'affichage de "Greater"

#include <stdio.h>

#include <stdlib.h>

void compute(int val) {

static int \_cnt = 0;

\_cnt += 2;

if (\_cnt > 4)

printf("Greater!\n");

}

int main() {

compute(1);

compute(2);

compute(3);

return 0;

}

21. Quelle est la longueur de la chaine thrdStr suivant le code ci-dessous ?

char newtStr[] = "SST-Campus\*&{([|$\*\*";

char \*thrdStr = malloc (sizeof(char)\*10);

strncpy(thrdStr,newtStr, strlen(thrdStr));

printf("strlen thrdStr (malloc by copy)=%d\n",strlen(thrdStr));

22. Quelle est la taille en mémoire du tableau arr ?

char arr[10] = {'a', 'b', 'c', '\0'};

Quelle est la longueur de la chaine secStr suivant le code ci-dessous ?

char newtabStr[] = "SST-Campus\*&{([|$\*\*";

char secStr[12];

strncpy(secStr,newtabStr, sizeof(secStr));

printf("strlen secStr ([12] by copy)=%d\n",strlen(secStr));

23. Donner l'affichage du programme C suivant

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main() {

int \*p;

int i = 10;

i++;

\*p = i;

printf("%d\n",i);

return 0;

}

24. Donner l'affichage du programme C suivant

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main() {

int i = 10;

int j = 11;

int \*p = 12;

\*p = i+j;

printf("%d\n",\*p);

return 0;

}

25. Donner l'affichage du programme C suivant

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main() {

int i = 10;

i++;

int \*p = &i;

i+= 15;

printf("%d\n",\*p);

return 0;

}

26. Ecrire la valeur résultante de 'd' :

//Ox désigne un nombre écrit en Héxa

int a = 0X8;

int b = 2, c = 4;

int d = a & b & c ;

27. Ecrire la valeur affichée :

//Ox désigne l'ecriture d'un nombre Héxa

//"%d" désigne l'affichage d'un nombre en décimal

int a = 0x018;

int b = 0x10;

printf("%d",a&b);

28. Ecrire la valeur affichée :

//Ox désigne l'ecriture d'un nombre Héxa

//"%d" désigne l'affichage d'un nombre en décimal

int a = 0x16;

int b = 6;

int c = 0x10;

printf("%d",a|b&c);

29. Ecrire la valeur affichée :

//Ox désigne l'ecriture d'un nombre Héxa

//%X permet l'affcihage d'un nombre en Héxa

int a = 25;

int b = 0x25;

printf("%X",a+b);

30. Ecrire la valeur affichée :

//Ox désigne l'ecriture d'un nombre Héxa

//%X permet l'affcihage d'un nombre en Héxa

int a = 0xABCDEF;

a++;a++;

printf("%X",a);