Eleito = AZUL

J = VERMELHO

0 1 2 3 4 5

U M D O I S

void InsertionSort(char vet[6], int tam=6) {

int i, j;

char eleito;

for (i = 1; i < tam(6); i++){

eleito(M) = vet[i](1);

j (0) = i - 1;

while ((j>=0) && (eleito < vet[j] (0) )){

vet[j+1] (1) = vet[j] (0);

j--(-1);

}

vet[j+1] (0) = eleito;

}

}

---------------------------------------------------------------------------------------------------------------------------------------------------------------

Eleito = AZUL

J = VERMELHO

0 1 2 3 4 5

M U D O I S

M U U O I S

D M U O I S

void InsertionSort(char vet[6], int tam=6) {

int i, j;

char eleito;

for (i = 2; i < tam(6); i++){

eleito(D) = vet[i](2);

j (1) = i - 1;

while ((j>=0) && (eleito < vet[j] (1) (0) )){

vet[j+1] (1) = vet[j] (0);

j--(-1);

}

vet[j+1] (0) = eleito;

}

}

---------------------------------------------------------------------------------------------------------------------------------------------------------------

Eleito = AZUL

J = VERMELHO

0 1 2 3 4 5

D M U O I S

D M U O I S

D M O U I S

void InsertionSort(char vet[6], int tam=6) {

int i, j;

char eleito;

for (i = 3; i < tam(6); i++){

eleito(O) = vet[i](3);

j (2) = i - 1;

while ((j>=0) && (eleito < vet[j] (2) (1) )){

vet[j+1] (3) = vet[j] (2);

j--(1);

}

vet[j+1] (2) = eleito;

}

}

---------------------------------------------------------------------------------------------------------------------------------------------------------------

Eleito = AZUL

J = VERMELHO

0 1 2 3 4 5

D M O U I S

D M O U I S

D M O U U S

D M O O U S

D M M O U S

D I M O U S

void InsertionSort(char vet[6], int tam=6) {

int i, j;

char eleito;

for (i = 4; i < tam(6); i++){

eleito(I) = vet[i](4);

j (3) = i - 1;

while ((j>=0) && (eleito < vet[j] (3) (2) (1) (0))){

vet[j+1] (4) (3) (2) = vet[j] (3)(2)(1);

j--(2)(1)(0);

}

vet[j+1] (1) = eleito;

}

}

---------------------------------------------------------------------------------------------------------------------------------------------------------------

Eleito = AZUL

J = VERMELHO

0 1 2 3 4 5

D I M O U S

D I M O U S

D I M O S U

void InsertionSort(char vet[6], int tam=6) {

int i, j;

char eleito;

for (i = 5; i < tam(6); i++){

eleito(S) = vet[i](5);

j (4) = i - 1;

while ((j>=0) && (eleito < vet[j] (4) (3) )){

vet[j+1] (5) = vet[j] (4);

j--(3);

}

vet[j+1] (4) = eleito;

}

}

---------------------------------------------------------------------------------------------------------------------------------------------------------------

Após organizar ficará

0 1 2 3 4 5

D I M O S U