** define CANT-INS 3 ** typology char leagunge [10];  ** define CANT-INS 3 ** typology char leagunge [10];  ** struct Noba 5  Int formularia [13];  Noba 6 syste;  3.  Int mores Br Categoria [cant act][cant ins][cant len]; / more 2 3D  Leagunge viengooges [cant len]; / vector con los nombres de los leagunges  Noba 6 histotimularias = NULL; / liste et la que supargo reciba los datas  Void Cargar formularias (Noba 6 testa hormularias ; int more she lategoria [fform trus] from e  Noba 6 aux = lista formularias;  While (Dux != NULL) 2  Intra actinidad = xx(aux > formularia) (1, 0, 2);  Intra leagunge = xx (aux > formularia) [1, 3, 5);  Intra leagunge = xx (aux > formularia) [1, 2, 2);  mores for lategoria [autinidad [finstitulian]] leagunge] + t;  aux = aux -> syre;  3  3  4 fracción xx, dado un vector, un micio x un fixal, fevuelve la  Bsiciar del vector (detro le ese micio y ese final) que contreve un  número distinto de cero. Es mu y util fora el poblema forque  Te ayula a saber que operón fire se cele cciana da en condo	int formulario [13];  Noto * syte;  int morros for Catagoria [cant Act][cant INS][cant Len]; /morriz 3D  lenguage vlenguages [cant Len]; / lector con los nambres de los lenguages  Noto * listatorinalarios = NULL; / lista en la que supargo recibo los lates  Noto * aux = listatorinalarios; (Noto * lista formularios , int mores for lategoría [[funt mis][funt Len]  Noto * aux = listatorinalarios;  While (bux = NULL) &  Int actividad = xx(aux - formularios), 0, 2);  Int lenguage = xx (aux - formularios [1, 0, 2);  Int lenguage = xx (aux - formularios [1, a, 12);  morris for lategoria [actividad] [ lassitucion] [ lenguage] + t;  aux = aux - sapet i  a finación xx / dado un vector, un unico y un final, feurelive la  ición del vector (dintro le ese mico y ese final) que contiene un  simero distinto le cero. Es muy util para el problima forque  ayuda a saber que opción fine cele ccon da es andor	# define CANT-INS 3	
Struct Nobo {  Noto * Softe;  The mores be lategoria [cart act][cart ins][cart len]; / manz 3D  Lenguage vergoges [cart len]; / rector con los nombres de los lenguages  Nodo * heratornoloxios = NVLL; / lista en la que supergo etcho los datos  Nodo * aux = lista formularios; int mores be lategoria [[fint maj] fourte  Nodo * aux = lista formularios;  While (aux = nVLL) {  Int actividad = xx (aux > formularios; int mores be lategoria [[fint maj] fourte  Int not rector xx (aux > formularios [], 6, 12);  Int lenguage = xx (aux > formularios [], 6, 12);  mores for lategoria [actividad ] finstitucion [] tenguage] ++;  aux = aux -> soft i  Bisciai del vector (sinto le ese mino y ese final) que contrele un  número distinto le cero. Es muy un final que contrele un  número distinto le cero. Es muy un final que contrele un	Int Motos Ber Categoria [CANT ACT][CANT INS][CANT LEN]; / Motors Cat Los Leaguages  Modo * Systateriolarios = NULL; // Ista en la que supergo etcho las Lates  Modo * Listateriolarios (Nodo * Ista formularios, int moloso la lategoria [[GANT Mas] GANT LEN  Nodo * aux = Istateriolarios;  While (Dux != NULL) &  Int actividad = xx (aux - Formulario [], 0, 2);  Int user tricion = xx (aux - Formulario [], 6, 12);  Int leaguage = xx (aux - Formulario [], 6, 12);  Int leaguage = xx (aux - Formulario [], 6, 12);  moloso for Categoria [actividad] [Institucion] [Leaguage] + +;  aux = aux - sapet;  3  3  4  6  6  6  7  7  7  7  7  7  7  7  7  7		1.
Int mores be caregola [cart act][cart ins][cart len]; / more 3D  Lenguage vlenguages [cart len]; / rector con los nambres de los lenguages  Nodo * Instrumentos (Nodo * histo hombories , int invorso los lategoría [ffent mos] formal  Nodo * aux = Instrumentos (Nodo * histo hombories , int invorso los lategoría [ffent mos] formal  Nodo * aux = Instrumentos (Nodo * histo hombories , int invorso los lategoría [ffent mos] formal  Nodo * aux = Instrumentos (Nodo * histo hombories , int invorso los lategoría [ffent mos] formal  Nodo * aux = Instrumentos (Nodo * histo hombories , int invorso los lategoría [ffent mos] formal  Nodo * aux = los ta hormbories ;  While (bux != NULL) &  Int actividad = xx (aux > formulario [1], 0, 2);  Int lenguage = xx (aux > formulario [1], 3, 5);  Int lenguage = xx (aux > formulario [1], 6, 12);  movios Ro (artegoria [actividad [finstitucion]] lenguage] ++;  aux = aux > sgre ;  da función xx, dado un vector, un visio y un fixal, devuelve la  Bosición del vector (ditto de ese nico y ele piral) que contrete un  número distinto de ceno. Es nuy vitil para el problema porque  Te ayuda a saber que opción fire celecciona da es anda	Int moros be Caregoria [cart act] [cant ins] [cant Len]; / morn2 3D  Longuage vlonguages [cant Len]; // rector con los nombres de les lesquages  Nodo * Institutularias = NVLL; // Issa en la que supargo recibio los datos  Nodo * aux = issa bomularias; int moros les Caregoria [[first mas] formates  Nodo * aux = issa bomularias;  While (bux! = NVLL) ?  Int actividad = xx(aux > formularia)[, 0, 3);  Int lenguage = xx (aux > formularia)[, 3, 5);  Int lenguage = xx (aux > formularia)[, 3, 12);  moros for Caregoria [actividad] [institucion][ lenguage] ++;  aux = aux -> syte;  3  A función xx, dado un vector, un micio x un firal, devielve la icián del vector (distro de ese micio x ese final) que contiere un finero distinto de cero. Es mux vitil para el problema porque ayuda a saber que opción fue se le cocona da en codor	Define CANT-LEN 7 (Proof Crust Congo (18)	
Int mores be caregola [cart act][cart ins][cart len]; / more 3D  Lenguage vlenguages [cart len]; / rector con los nambres de los lenguages  Nodo * Instrumentos (Nodo * histo hombories , int invorso los lategoría [ffent mos] formal  Nodo * aux = Instrumentos (Nodo * histo hombories , int invorso los lategoría [ffent mos] formal  Nodo * aux = Instrumentos (Nodo * histo hombories , int invorso los lategoría [ffent mos] formal  Nodo * aux = Instrumentos (Nodo * histo hombories , int invorso los lategoría [ffent mos] formal  Nodo * aux = Instrumentos (Nodo * histo hombories , int invorso los lategoría [ffent mos] formal  Nodo * aux = los ta hormbories ;  While (bux != NULL) &  Int actividad = xx (aux > formulario [1], 0, 2);  Int lenguage = xx (aux > formulario [1], 3, 5);  Int lenguage = xx (aux > formulario [1], 6, 12);  movios Ro (artegoria [actividad [finstitucion]] lenguage] ++;  aux = aux > sgre ;  da función xx, dado un vector, un visio y un fixal, devuelve la  Bosición del vector (ditto de ese nico y ele piral) que contrete un  número distinto de ceno. Es nuy vitil para el problema porque  Te ayuda a saber que opción fire celecciona da es anda	Int moros be Caregoria [cart act][cart ins][cart len]; / moroz 3D  Longuage vlonguages [cart len]; // rector con los nameros la los lesquages  Nodo * Institutularios = NVLL; // Issa en la que supargo recibo los datos  Ordo cargas formularios (Nalo * insta formularios, int moros for lategoria [[first mos] formatea  Nodo * aux = issta formularios;  While (bux! = NVLL) ?  Int actividad = xx(aux > formulario[1, 0, 2);  Int lenguage = xx (aux > formulario[1, 3, 5);  Int lenguage = xx (aux > formulario[1, 3, 5);  Int lenguage = xx (aux > formulario[1, 3, 12);  moros for Cartegoria [actividad] [institucion][lenguage] ++;  aux = aux -> syte;  3  A formular xx, dado un vector, un inicio x un final, devuelve la  icián del vector (distro de ese inicio x ese final) que contiene un  smero distinto de cero. Es mux util para el problema porque  ayuda a saber que opción fine selecciona da en ando	Struct Mal-C	
Noto * syte;  Int mlores Br Categoria [cart AcT][cart INS][cart Len]; / morniz 3D  Lenguage vlenguages [cart Len]; / rector con los nambers de los lenguages  Nodo * histotimolorias = NULL; // lista en la que supargo reciso los lates  Void Corgar Brancharios (Nobo * histo homorius , int mlotos be lategoria [Hant ws] [cart Lenguage   Nobo * aux = lista formularios;  While (aux = NULL) &  Int actividad = xx (aux > formulario [1, 0, 2);  Int lenguage = xx (aux > formulario [1, 3, 5);  Int lenguage = xx (aux > formulario [1, 4, 12);  mloros Par Categoria [actividad] [listitucios] [language] ++;  aux = aux -> syrt ;  3  Biscian del vector (dutas de ese mins x ese pinal) que contieve un número distinto de cero. Es mux Util para el problema parque  Te ayuda a saber que apeió fire seleccionada en cardar	Noto * syte;  Int moors or Categoria [CANT ACT][CANT INS][CANT LÉN]; / moors 3D  Longuage vierguages [CANT LEN]; // vector con los nambres de los lenguages  Nodo * Instatomolorios = NVLL; // lista en la que supargo recino los datos  Ond Cargar formularios (Nodo * hista formularios , int moors for Categoria [[Fant we]] Fent Lea  Nodo * aux = lista formularios;  While (aux = NVLL) &  Int actividad = xx (aux - formulario [], 0, 2),  Int lenguage = xx (aux - formulario [], 3, 5);  Int lenguage = xx (aux - formulario [], 6, 12);  moors for Categoria [actividad] [Institución] [lenguage] ++;  aux = aux -> syte;  3  4  Anció xx, dado un vector, un micio y un final, devivelve la  icián del vector (datos de ese micio y ese final) que contreve un  imbro distinto de cero. Es muy oful para el problema porque  aquida a saber que opeión fire cele carra da en codar		
Noto * syte;  Int mlores Br Categoria [cart AcT][cart INS][cart Len]; / morniz 3D  Lenguage vlenguages [cart Len]; / rector con los nambers de los lenguages  Nodo * histotimolorias = NULL; // lista en la que supargo reciso los lates  Void Corgar Brancharios (Nobo * histo homorius , int mlotos be lategoria [Hant ws] [cart Lenguage   Nobo * aux = lista formularios;  While (aux = NULL) &  Int actividad = xx (aux > formulario [1, 0, 2);  Int lenguage = xx (aux > formulario [1, 3, 5);  Int lenguage = xx (aux > formulario [1, 4, 12);  mloros Par Categoria [actividad] [listitucios] [language] ++;  aux = aux -> syrt ;  3  Biscian del vector (dutas de ese mins x ese pinal) que contieve un número distinto de cero. Es mux Util para el problema parque  Te ayuda a saber que apeió fire seleccionada en cardar	Noto * syte;  Int moors or Categoria [CANT ACT][CANT INS][CANT LÉN]; / moors 3D  Longuage vierguages [CANT LEN]; // vector con los nambres de los lenguages  Nodo * Instatomolorios = NVLL; // lista en la que supargo recino los datos  Ond Cargar formularios (Nodo * hista formularios , int moors for Categoria [[Fant we]] Fent Lea  Nodo * aux = lista formularios;  While (aux = NVLL) &  Int actividad = xx (aux - formulario [], 0, 2),  Int lenguage = xx (aux - formulario [], 3, 5);  Int lenguage = xx (aux - formulario [], 6, 12);  moors for Categoria [actividad] [Institución] [lenguage] ++;  aux = aux -> syte;  3  4  Anció xx, dado un vector, un micio y un final, devivelve la  icián del vector (datos de ese micio y ese final) que contreve un  imbro distinto de cero. Es muy oful para el problema porque  aquida a saber que opeión fire cele carra da en codar	Int formularing to 27:	
Int mores by Caregoria [cant acti] [cant ins] [cant Len]; / more 3D  Lenguage vienguages [cant Len]; / vector con los nambres de los lenguages  Nodo * Instatornularios = NULL; // insta en la que supargo recibo los dates  Void Cargar formularios (Nodo * lista formularios , int mores la lategoria [Henne-ms] [cant lenguage] **  Nodo * aux = lista formularios;  While (aux != NULL) &  (Int actividad = xx (aux > formulario [1, 0, 2);  Int most recipo = xx (aux > formulario [1, 0, 2);  Int lenguage = xx (aux > formulario [1, a, 12);  moloros for Cartegoria [actividad] [institucion] [lenguage] ++;  aux = aux -> sqte;  3  Bosició del vector (distro le ese mino y un qual, devielve la  número distinto de cero. Es muy vitil para el problema porque  Te ayuda a saber que operió fire cele cciona da en cirdar	Int mores Br Caregoria [carr act][cart ins][cart Lén]; /morns 3D  Lenguage vienguages [cart Lén]; /vector con los nambres de los legiones.  Modo * Institutularios = NULL; //lista en la que supargo reciso los datos  loid Cargar Formularios (Nobo * Ista Formularios , int invorsala Caregoria [Hant ins][cart lea  Nodo * aux = lista Formularios;  While (aux != NULL) &  Int actividad = xx (aux -> Formulario[1, 0, 2);  Int institución = xx (aux -> Formulario[1], a, 12);  Int lenguage = xx (aux -> Formulario[1], a, 12);  milosos Par Cartegoria [actividad][Institución][Lenguage] ++;  aux -> cyte;  3  Formulario [1], a, 12);  milosos Par Cartegoria [actividad][Institución][Lenguage] ++;  aux -> cyte;  3  Formulario [1], a, 12);  milosos Par dado un vector, un micio y un fical, devivelve la  icián del vector (dintro de ese micio y ese final) que contreve un  imbro distinto de cero. Es muy vitil para el problema porque  avuda a saber que opená fine cele carar da el cadar		
Int mores by Caregoria [cant acti] [cant ins] [cant Len]; / more 3D  Lenguage vienguages [cant Len]; / vector con los nambres de los lenguages  Nodo * Instatornularios = NULL; // insta en la que supargo recibo los dates  Void Cargar formularios (Nodo * lista formularios , int mores la lategoria [Henne-ms] [cant lenguage] **  Nodo * aux = lista formularios;  While (aux != NULL) &  (Int actividad = xx (aux > formulario [1, 0, 2);  Int most recipo = xx (aux > formulario [1, 0, 2);  Int lenguage = xx (aux > formulario [1, a, 12);  moloros for Cartegoria [actividad] [institucion] [lenguage] ++;  aux = aux -> sqte;  3  Bosició del vector (distro le ese mino y un qual, devielve la  número distinto de cero. Es muy vitil para el problema porque  Te ayuda a saber que operió fire cele cciona da en cirdar	Int mores Br Caregoria [carr act][cart ins][cart lén]; /morn2 3D  Lenguage vienguages [cart lén]; /vector con los nambres de los lenguages  Nodo * Institutularios = NVLL; //Ista en la que supergo reciso los datos  lord Cargar Formularios (Nobo * Insta Formularios , int mores la Caregoria [Hant ins][cart lenguage in la gue supergo reciso los datos  Nodo * aux = Insta Formularios;  While (aux != NVLL) &  (Int actividad = xx (aux -> Formulario[1], 0, 2);  (Int lenguage = xx (aux -> Formulario[1], 0, 12);  (Int lenguage = xx (aux -> Formulario[1], 1, 2, 12);  mores Par Cartegoria [actividad][Institucion][lenguage] ++;  aux = aux -> syte;  3  4  Formulario [1], 10, 11, 12, 12, 13, 13, 13, 13, 13, 13, 13, 13, 13, 13	Non * sate:	
Int mores by Caregoria [cart act] [cart ins] [cart len]; / mores 30  Lenguage vlenguages [cart len]; / rector con los nombres de los lenguages  Nodo * listatornolorios = NULL; // lista en la que supargo recibo los datos  Void Cargar formularios (Nodo * lista formularios , int mores la lategoria [ffinit ms] fenite  Nodo * aux = lista formularios;  While (aux != NULL) &  (Int actindad = xx (aux > formulario [1, 0, 2);  Int lenguage = xx (aux > formulario [1, 3, 5);  Int lenguage = xx (aux > formulario [1, 6, 12);  mores for lategoria [autinidad] [institucion] [ enguage] ++;  aux = aux -> sqre i  Bsición del vector (dintro de ese mino y ese final) que contreve um  número distinto de cero. Es muy vitil para el problema porque  Te ayuda a saber que opción fine selecciona da en cridar	Int mores be Categoría [cant aci][cant ins][cant len]; / morn 2 3D  lenguage vlenguages [cant len]; / vector con los nombres de los lenguages  Modo * histatornolonios = NULL; // lista en la que supargo recibo los dartos  loid corgan formularios (Nobo * hista formularios , int mores la lategoría [][cant wo][cant len  Nodo * aux = lista formularios;  While (aux = NULL) &  Int actividad = xx (aux = formularios [], 0, 2);  Int lenguage = xx (aux = formularios [], 3, 5);  Int lenguage = xx (aux = formularios [], a, 2);  mores la lategoría [autividad [[Institución][lenguage]] ++;  aux = aux = sigte i  a formular de la vector (dintro de ese mico y un fical) fevuelve la  icián del vector (dintro de ese mico y ese final) que contrele um  imbro distinto de cero. Es muy uni para el problema porque  oyuda a saber que opción fine cele cciona da en cirdar	7.	
Lenguage Vlengoages [CANT LEN]; // vector car los nambres de los lenguages  Nodo * listatornularios = NVLL; // lista en la que supergo recibio las datos  Void cargar formularios (Nodo * lista formularios , int milotos la lategoría [HEMMET INS] forme la Nodo * aux = lista formularios;  While (aux != NVLL) &  Int astronad = xx (aux > formularios [], a, z);  Int lenguage = xx (aux > formularios [], a, z);  Int lenguage = xx (aux > formularios [], a, z);  milotos la Catgaria [attribad] [institucion] [lenguage] ++;  aux = aux -> syre;  aux = aux -> syre;  la función xx, dado un vector, un micio y un final, devivelve la la la función del vector (dintro le ese micio y ese pinal) que contreve un número distinto de cero. Es mu y vitil para el problema parque  TC algular a saber que opción fine celecciona da en andar	Lenguage vlenguages [cant LEN]; // vector con los nombres de los lenguages  Nodo * Instatornularios = NULL; // lista en la que supargo recisa las datos  Old Cargar formularios (Nodo * lista formularios , int molosola lategoría [][[innt-ms][[innt-ens][[i	5'	
Lenguage Vlengoages [CANT LEN]; // vector car los nambres de los lenguages  Nodo * listatornularios = NVLL; // lista en la que supergo recibio las datos  Void cargar formularios (Nodo * lista formularios , int milotos la lategoría [HEMMET INS] forme la Nodo * aux = lista formularios;  While (aux != NVLL) &  Int astronad = xx (aux > formularios [], a, z);  Int lenguage = xx (aux > formularios [], a, z);  Int lenguage = xx (aux > formularios [], a, z);  milotos la Catgaria [attribad] [institucion] [lenguage] ++;  aux = aux -> syre;  aux = aux -> syre;  la función xx, dado un vector, un micio y un final, devivelve la la la función del vector (dintro le ese micio y ese pinal) que contreve un número distinto de cero. Es mu y vitil para el problema parque  TC algular a saber que opción fine celecciona da en andar	Lenguage vlenguages [cant LEN]; // vector con los nombres de los lenguages  Nodo * Instatornularios = NULL; // lista en la que supargo recisa las datos  Old Cargar formularios (Nodo * lista formularios , int molosola lategoría [][[innt-ms][[innt-ens][[i		
Lenguage Vlengoages [CANT LEN]; // vector car los nambres de los lenguages  Nodo * listatornularios = NVLL; // lista en la que supergo recibio las datos  Void cargar formularios (Nodo * lista formularios , int milotos la lategoría [HEMMET INS] forme la Nodo * aux = lista formularios;  While (aux != NVLL) &  Int astronad = xx (aux > formularios [], a, z);  Int lenguage = xx (aux > formularios [], a, z);  Int lenguage = xx (aux > formularios [], a, z);  milotos la Catgaria [attribad] [institucion] [lenguage] ++;  aux = aux -> syre;  aux = aux -> syre;  la función xx, dado un vector, un micio y un final, devivelve la la la función del vector (dintro le ese micio y ese pinal) que contreve un número distinto de cero. Es mu y vitil para el problema parque  TC algular a saber que opción fine celecciona da en andar	Lenguage vlenguages [cant LEN]; // vector con los nombres de los lenguages  Nodo * Instatornularios = NULL; // lista en la que supargo recisa las datos  Old Cargar formularios (Nodo * lista formularios , int molosola lategoría [][[innt-ms][[innt-ens][[i	INT myotos for lategorica [CANT_ACT] [CANT_INS] [CANT_LEN]; / MOTOS 3D	
Note   State intension   Null;   It ista la que supergo recita la catas  Void cargor formularios (Note * Ista formularios , int milotos la lategoría [][[mit mos][[cmit mos][[cm	Nodo * histatornularios = NULL; // lista en la que supargo recibo las datos  lord congartornularios (Nodo * lista hornularios , int mlotos la lategoría [] fent ensistentes  Nodo * aux = lista hornularios;  While (aux! = NULL) &  Int actividad = xx (aux > formulario[1, 0, 2);  Int instritucion = xx (aux > formulario[1, 3, 5);  Int lenguage = xx (aux > formulario [1], a, 12);  mlotos Par Categoria [actividad] [Institucion] Lenguage] ++;  aux = scyte;  a función xx, dado un vector, un inicio y un final, devuelve la  ición del vector (dintro de ese micio y ese final) que contiere un  imero distinto de cero. Es muy vitil para el problema porque  ovueda a saber que operón fine se le cciona da en order		
Nodo * Instatornalarios - NVIL; // Insta la la que supergo reciso los catos.  Void corgorformalatios (Nodo * Ista formalarios, int milotos la lategoría [][first=ins]	Modo * listatornoloriós = NULL; // lista en la que supergo reciso los datos  Ordo Corgor formularios (Nalo * lista formularios , int molora la lategoría [][[int] mas][[int] Len  Nodo * aux = lista formularios;  While (aux != NULL) &  Int actividad = xx (aux > formulario [], 0, 2),  Int institución = xx (aux > formulario [], 3, 5);  Int lenguage = xx (aux > formulario [], 6, 12);  moloros for Categoría (actividad ][Institución][Lenguage] ++;  aux = aux -> syte;  3  A forción xx , dado un vector, un inicio y un final, devuelve la  ición del vector (dintro le ese micio y ese final) que contiere un  imero distinto de cero. Es muy vitil para el problema forque  ovueda a saber que opción fine se le cciona da en conder	Lenguage VLenguages [CANT LEN]; / Vector con los nambres de los lenguages	-
Void Corgar for mularios (Nobo * hista formborios, int moloso la lategoría [] fint = mis] fint = 1 Nodo * aux = lista formborios;  While (aux! = NULL) &  (Int actividad = xx (aux > formborio [], 0, 2);  Int institución = xx (aux > formborio [], 3, 5);  Int lenguage = xx (aux > formborio [], 6, 12);  molosos la lategoría [actividad] [institución] (lenguage] ++;  aux = aux -> sigre;  3  3  4  Bisición del vector (dintro de ese uición y ese pina) que connece un número distinto de cero. Es nu y un para el problema porque  TC aluda a saber que opción pre seleccionada en condin	bold Cargar for Aulatios (Nobo * Ista formbrios, int motos for Categoria [Hant-Ws] [CANT-LEAN Nobo * aux = Ista formbrios;  While (Dux != NULL) &  Int actividad = xx (aux > Formbrio[], 0, 2);  Int Instruction xx (aux > Formbrio[], 3, 5);  Int lenguage = xx (aux > Formbrio[], 6, 12);  motos for Categoria [actividad] [Institucion] (benguage] ++;  aux = aux -> syte;  a función xx, dado un vector, un inicio y un final, devirelve la icián del vector (dintro de ese unico y ese final) que connere un finero distinto de cero. Es muy util para el problema porque aquela a sober que opción fine cele caona da en anda		
Void cargar formularios (Nobo* hista formularios, int molota la lategoría [][[mt ms][[amt ms]][amt    Nodo * aux =   1sta formularios;  While (aux != NULL) &  Int actividad = xx (aux > formulario [], 0, 2);  Int 10st itacion = xx (aux > formulario [], 3, 5);  Int lenguage = xx (aux > formulario [], 8, 12);  molota Par Categoria [autividad] [institucion][lenguage] ++;  aux = aux -> syte;  3  3  4  Basición del vector (dutro de ese mino y un final, devuelve la  Basición del vector (dutro de ese mino y ese punal) que connece un  númbro distinto de cero. Es muy otil para el problema parque  Te ayuda a saber que opción fine cele caona da en andir	Nodo * aux = lista formularios;  While (aux != NULL) &  Int actividad = xx (aux > formularios [], 0, 2);  Int institución= xx (aux > formularios [], 3, 5);  Int lenguage = xx (aux > formularios [], 6, 12);  Motos Por Cartagoria (actividad ] finstitución } ( Lenguage ] ++;  aux = aux -> syre;  a función xx, dado un vector, un inicio y un final, devivelve la ición del vector (dintro de ese micio y ese final) que contreve un impro distinto de cero. Es muy util para el problema porque ayuda a saber que opción fre se le caona da en cordar	110do - listatornularios = 1446, 1/1sta en la que suporgo reciso los das	105
Void cargar formularios (Nobo* hista formularios, int molota la lategoría [][[mt ms][[amt ms]][amt    Nodo * aux =   1sta formularios;  While (aux != NULL) &  Int actividad = xx (aux > formulario [], 0, 2);  Int 10st itacion = xx (aux > formulario [], 3, 5);  Int lenguage = xx (aux > formulario [], 8, 12);  molota Par Categoria [autividad] [institucion][lenguage] ++;  aux = aux -> syte;  3  3  4  Basición del vector (dutro de ese mino y un final, devuelve la  Basición del vector (dutro de ese mino y ese punal) que connece un  númbro distinto de cero. Es muy otil para el problema parque  Te ayuda a saber que opción fine cele caona da en andir	Nodo * aux = lista formularios;  While (aux != NULL) {  Int actividad = xx (aux > formulario[], 0, 2);  Int lesgrage = xx (aux > formulario[], 3, 5);  Int lesgrage = xx (aux > formulario[], 6, 12);  Motos Por Cartagoria [actividad][institucion][lesgrage] ++;  aux = aux -> syre;  a función xx, dado un vector, un micio y un fixal, devivelve la  ición del vector (distrib de ese micio y ese final) que contreve un  imero distinto de cero. Es muy util para el problema porque  ayuda a saber que opción fixe se le caona da es cridin		+
Void cargar formularios (Nobo* hista formularios, int molota la lategoría [][[mt ms][[amt ms]][amt    Nodo * aux =   1sta formularios;  While (aux != NULL) &  Int actividad = xx (aux > formulario [], 0, 2);  Int 10st itacion = xx (aux > formulario [], 3, 5);  Int lenguage = xx (aux > formulario [], 8, 12);  molota Par Categoria [autividad] [institucion][lenguage] ++;  aux = aux -> syte;  3  3  4  Basición del vector (dutro de ese mino y un final, devuelve la  Basición del vector (dutro de ese mino y ese punal) que connece un  númbro distinto de cero. Es muy otil para el problema parque  Te ayuda a saber que opción fine cele caona da en andir	Nodo * aux = lista formbrios;  While (aux != NULL) {  Int actividad = xx (aux > formbrio[], 0, 2);  Int lesgrage = xx (aux > formbrio[], a, 12);  Int lesgrage = xx (aux > formbrio[], a, 12);  Intotos Por Cartagoria [actividad][institucion][lesgrage] ++;  aux = aux -> syre;  a función xx, dado un vector, un micio y un fixal, fevuelve la  ición del vector (distro de ese micio y ese final) que contreve un  imero distinto de cero. Es muy util para el problema porque  ayuda a saber que opción fixe se le caonada en cricia		$\vdash$
Nodo * aux = lista formbrios;  While (aux!=NULL) {  Int actividad = xx (aux > formbrio[], 0, 2);  Int not itain xx (aux > formbrio[], 3, 5);  Int lenguage = xx (aux > formbrio[], 8, 12);  molosos for lategoria [actividad][institución][enguage] ++;  aux = aux -> sqre;  {  Busición del vector (distro de ese micio y un final, fevrelve la lessición del vector (distro de ese micio y ese final) que contreve un número distinto de cero. Es muy util para el poblera porque  Te ayuda a saber que opción fine selecciona da es conditions	Nodo * aux = lista formbrios;  While (aux! = NULL) {  Int actividad = xx (aux > formbrio[], 0, 2);  Int lesgrage = xx (aux > formbrio[], 3, 5);  Int lesgrage = xx (aux > formbrio[], 6, 12);  Motos for lategoria [actividad][institucion][lenguage] ++;  aux = aux -> syre;  a función xx, dado un vector, un micio y un fixal, fevuelve la  ición del vector (distribute ese micio y ese final) que contreve un  imero distinto de cero. Es muy vitil para el problema porque  ayuda a saber que opción fixe se le caonada en cricia	Void Cos as 6 say a sias (Note + lista formbosios int myotos for Categoria sassia sassia	
While (aux!=NULL) {  Int actividad = xx(aux > formulario[1, 0, 2);  Int 105t tracion = xx (aux > formulario[1, 3, 5);  Int lenguage = xx (aux > formulario[1, 6, 12);  Motos Por Categoria [actividad] [Institución] [Lenguage] ++;  aux = aux -> syte;  }  La función xx, dado un vector, un micio y un final, fervelve la  Bosición del vector (distro de ese micio y ese final) que contreve un  número distinto de cero. Es muy útil para el problema porque  Te ayuda a saber que opción five seleccionada es cada	While (aux!=NULL) &  (Int actividad = xx(aux > formbrio[], 0, 2);  (Int lenguage = xx (aux > formbrio[], a, 12);  (Int lenguage = xx (aux > formbrio[], a, 12);  (Int lenguage = xx (aux > formbrio[], a, 12);  (Int lenguage = xx (aux > formbrio [], a, 12);  (Int lenguage	THE COURT OF THE CONTROL OF THE CONT	1-Cer
While (aux!=NULL) &  Int actividad = xx (aux > formulario[], 0, 2);  Int 105t, tocon= xx (aux > formulario[], 3, 5);  Int lenguage = xx (aux > formulario[], 6, 12);  Moros Por Categoria [actividad] [Institución] Lenguage] ++;  aux = aux -> syrc;  3  Sición del vector (distro de ese mino y un final, devirelve da  BSICIÓN del vector (distro de ese mino y ese final) que contreve un  número distinto de cero. Es muy vitil para el problema porque  Te ayuda a saber que opción five se elecciona da en cadar	While (aux!=NULL) &  (Int actividad = xx(aux > formbrio[], 0, 2);  (Int lenguage = xx (aux > formbrio[], a, 12);  (Int lenguage = xx (aux > formbrio[], a, 12);  (Int lenguage = xx (aux > formbrio[], a, 12);  (Int lenguage = xx (aux > formbrio [], a, 12);  (Int lenguage	Nodo * aux = 15ta torneloriós:	
Int actividad = xx(aux - formulario [1, 0, 3);  10t 105t itacion = xx (aux -> formulario [1, 3, 5);  10t 10st itacion = xx (aux -> formulario [1, 3, 5);  10t 10st itacion = xx (aux -> formulario [1, 6, 12);  10t 10st itacion = xx (aux -> formulario [1, 6, 12);  10t 10st itacion = xx (aux -> formulario [1, 6, 12);  10t 10st itacion = xx (aux -> formulario [1, 0, 3);  10t 10st itacion = xx (aux -> formulario [1, 0, 12);  10t 10st itacion = xx (aux -> formulario [1, 0, 12);  10t 10st itacion = xx (aux -> formulario [1, 0, 12);  10t 10st itacion = xx (aux -> formulario [1, 0, 12);  10t 10st itacion = xx (aux -> formulario [1, 0, 12);  10t 10st itacion = xx (aux -> formulario [1, 0, 12);  10t 10st itacion = xx (aux -> formulario [1, 0, 12);  10t 10st itacion = xx (aux -> formulario [1, 0, 12);  10t 10st itacion = xx (aux	Int actividad = xx (aux - formbrio [], 0, 3);  Int 105+ tracion = xx (aux -> formbrio [], 3, 5);  Int lenguage = xx (aux -> formbrio [], 6, 12);  Motos Par lategoria [actividad] [institución] [enguage] ++;  aux = aux -> syre;  3  4  Anción xx, dado un vector, un micio y un final, devirelve la lición del vector (distro de ese micio y ese final) que contreve un finero distinto de cero. Es muy estil para el problema porque oquida a saber que opción fine selecciona da es corden		
Int actividad = xx (aux -> Formulario [1, 0, 3);  10t 105t itaion = xx (aux -> Formulario [1, 3, 5);  10t 10st itaion = xx (aux -> Formulario [1, 3, 5);  10t 10st itaion = xx (aux -> Formulario [1, 6, 12);  10t 10st language = xx (aux -> Formulario [1, 6, 12);  10t 10st language = xx (aux -> Formulario [1, 6, 12);  10t 10st language = xx (aux -> Formulario [1, 0, 3);  10t 10st language = xx (aux -> Formulario [1, 0, 3);  10t 10st language = xx (aux -> Formulario [1, 0, 3);  10t 10st language = xx (aux -> Formulario [1, 0, 3);  10t 10st language = xx (aux -> Formulario [1, 0, 3);  10t 10st language = xx (aux -> Formulario [1, 0, 3);  10t 10st language = xx (aux -> Formulario [1, 0, 3);  10t 10st language = xx (aux -> Formulario [1, 0, 3);  10t 10st language = xx (aux -> Formulario [1, 0, 3);  10t 10st language = xx (aux -> Formulario [1, 0, 3);  10t 10st language = xx (aux -> Formulario [1, 0, 3);  10t 10st language = xx (aux -> Formulario [1, 0, 3);  10t language = xx (aux -> Formulario [1, 0, 3);  10t language = xx (aux -> Formulario [1, 0, 12);  10t language = xx (aux -> Formul	Int actividad = xx (aux -> formbrio [], 0, 2);  Int 105+ tracion = xx (aux -> formbrio [], 3, 5);  Int lenguage = xx (aux -> formbrio [], 6, 12);  Motos Par Categoria [actividad] [institución] [lenguage] ++;  aux = aux -> syre;  3  4  4  4  4  4  4  4  4  4  4  4  4	While (aux != NULL) ?	
Int lenguage = xx (aux -> formulario [1, 3, 5);  Int lenguage = xx (aux -> formulario [1, 6, 12);  more Por Categoria (actividad) [Institución) [Lenguage] ++;  aux -> syte;  la función xx, dado un vector, un micio y un pinal, fevuelve la  Bosición del vector (distro de ese micio y ese pinal) que connece un  número distinto de cero. Es my útil para el problema porque  Te ayuda a saber que opción pre selecciona da en cirdar	Int 1054, tocon= xx (aux -> formulario [], 3, 5);  111+ lenguage = xx (aux -> formulario [], 6, 12);  111+ lenguage = xx (aux -> formulario [], 6, 12);  111+ lenguage = xx (aux -> formulario [], 6, 12);  111+ lenguage = xx (aux -> formulario [], 6, 12);  111+ lenguage = xx (aux -> formulario [], 6, 12);  111+ lenguage = xx (aux -> formulario [], 6, 12);  111+ lenguage = xx (aux -> formulario [], 6, 12);  111+ lenguage = xx (aux -> formulario [], 6, 12);  111+ lenguage = xx (aux -> formulario [], 6, 12);  111+ lenguage = xx (aux -> formulario [], 6, 12);  112+ l		
mvotos Par Categoria (actividad) [Institución] (Lenguage) ++;  aux -> sorte;  la función xx, dado un vector, un micio y un final, devielve la  Bosición del vector (dentro de ese micio y ese pinal) que connece un  númbro distinto de cero. Es muy útil para el problema porque  Te ovuda a saber que opción fine se le caora da en condur	moloros Por Cartegoria (actividad) [Institución] [Lenguage] ++;  aux = aux -> syre;  a función xx, dado un vector, un micio y un final, devivelve la  ición del vector (distro de ese micio y ese final) que contrere un  imero distinto de cero. Es muy útil para el problema porque  nyuda a saber que opción fine se se le capita da es cordin	Int activided = xx (aux -> formilario [], 0, 2);	
mvotos Par Categoria [actividad] [institución] [lenguage] ++;  aux -> syte;  3  La función xx, dado un vector, un micio y un final, devuelve la  BSICIÓN del vector (destro de ese micio y ese funal) que contreve un  número distinto de cero. Es muy útil para el problema porque  Te ayuda a saber que escuón fine se ese caora da en ardir	mlotos Par Categoria (actividad) [Institución) [Lenguage] ++;  OUX = aux -> syte;  a función xx, dado un vector, un micio y un final, devuelve la  ición del vector (dutro de ese micio y ese final) que contreve un  imbro distinto de cero. Es muy util para el problema porque  nyuda a sober que opción que se le caora da en ardor	10+ 105+16con= xx (aux -> formulario [], 3, 5);	+-
mvotos Par Categoria [actividad] [institución] [lenguage] ++;  aux -> syte;  3  La función xx, dado un vector, un micio y un final, devuelve la  BSICIÓN del vector (destro de ese micio y ese funal) que contreve un  número distinto de cero. Es muy útil para el problema porque  Te ayuda a saber que escuón fine se ese caora da en ardir	mlotos Par Categoria (actividad) [Institución) [Lenguage] ++;  OUX = aux -> syte;  a función xx, dado un vector, un micio y un final, devuelve la  ición del vector (dutro de ese micio y ese final) que contreve un  imbro distinto de cero. Es muy util para el problema porque  nyuda a sober que opción que se le caora da en ardor	Int lenguage = xx (aux -> formulario [], 6, 12),	+
OUX = aux -> syre;  3  La función xx, dado un vector, un micio y un final, devuelve la  BSICIÓN del vector (distro de ese micio y ese final) que connese un  Número distinto de cero. Es muy util para el problema porque  Te ayuda a saber qué opción que celecciona da en cirdor	a función xx, dado un vector, un micio y un final, fervelve la sición del vector (dintro de ese micio y ese final) que contrere un ímbro distinto de cero. Es muy útil para el problema porque aquela a saber que opción que celeccionada en cada		++
BSICIÓN DE VECTOR (distrio de ese micio y un final) que connere un número distinto de cero. Es muy útil para el problema porque Te ayuda a saber que opción fine se le caonada en anda	A función XX, dado un vector, un micio y un final, fervelve la ición del vector (dintro de ese micio y ese final) que contrere un ímbro distinto de cero. Es muy útil para el problema porque aque a saber que opción que celeccionada en order	mnotos for Untegora [actividad ][institution][lenguage] TT	
BSICIÓN DE VECTOR (distrio de ese micio y un final) que connere un número distinto de cero. Es muy útil para el problema porque Te ayuda a saber que opción fine se le caonada en anda	A función XX, dado un vector, un micio y un final, devuelve la ición del vector (dintro de ese micio y ese final) que contrere un ímbro distinto de cero. Es muy útil para el problema porque aque a saber que opción que celeccionada en circur	Olly - gux > Cua 4	
Residion del vector (distro de ese minio y ese pinal) que connere un número distinto de cero. Es muy util para el problema porque Te ajuda a sober que opción que seleccionada en cadar	ición del vector (dentro de ese micio y ese punal) que contrere un imero distinto de cero. Es muy vitil para el problema porque oque a sober que opción pre seleccionada en cadar		
Residion del vector (distro de ese minio y ese pinal) que connere un número distinto de cero. Es muy util para el problema porque Te ajuda a sober que opción que seleccionada en cadar	ición del vector (dentro de ese micio y ese punal) que contrere un imero distinto de cero. Es muy vitil para el problema porque oque a sober que opción pre seleccionada en cadar		
RSICIÓN del vector (distro de ese micio y ese pinal) que connere un número distinto de cero. Es muy util para el problema porque Te ayuda a saber que opción que seleccionada en cadar	ición del vector (dentro de ese micio y ese punal) que contrere un imero distinto de cero. Es muy vitil para el problema porque oque a sober que opción pre seleccionada en cadar		
RSICIÓN del vector (distro de ese micio y ese pinal) que connere un número distinto de cero. Es muy útil para el problema porque Te ayuda a saber que opción que seleccionada en cadar	ición del vector (dentro de ese micio y ese punal) que contrere un imero distinto de cero. Es muy vitil para el problema porque oque a sober que opción que celeccionada en cadar		
RSICIÓN del vector (distro de ese micio y ese pinal) que connere un número distinto de cero. Es muy util para el problema porque Te ayuda a saber que opción que seleccionada en cadar	ición del vector (dentro de ese micio y ese punal) que contrere un imero distinto de cero. Es muy vitil para el problema porque oque a sober que opción pre seleccionada en cadar		
RSICIÓN del vector (distro de ese micio y ese pinal) que connere un número distinto de cero. Es muy útil para el problema porque Te ayuda a saber que opción que seleccionada en cadar	ición del vector (destro de ese micio y ese pinal) que contrere un imero distinto de cero. Es muy vill para el problema porque apuda a sober que opción que celeccionada en cadar	la función XX. dado un vector un micio y un final bruselve la	
número distinto le cero. Es muy vitil para el problema porque	ímero distinto de cero. Es muy vitil para el problema porque aque a saber que opción que seleccionada en circu		
número distinto de cero. Es muy vitil para el problema porque re ayuda a saber qué opción pre seleccionada en circu	ímero distinto de cero. Es muy vitil para el problema porque aque a saber que opción que seleccionada en circu	Posición del vector (destro de ese micio y ese pina) que contrere un	
re ayuda a saber que opción que seleccionada en circa	ayuda a saber qué opción que seleccionada en circo		
re ayuda a saber que opción que seleccionada en circa	ayuda a saber qué opción que seleccionada en circo	NUMBRO distinto de cero. Es muy util para el problema porque	re
		The ayuda a sober que opción que seleccionada en circur	
Catégorios de un formibirio.	tegorios de un formitario.		11
		Catégorios de un pormitario.	-
			-
			-
<del>                                      </del>			
			-

I was to start of the Void mostros Resultadas (leguaje vleyvajes [], int mloros la lategoria [][=][6], int act, int 1/15+) { INT VOTOS [CANTLEN] {}; doliga a inicializar el vector int summitoria =0; int alumnos =0; int total =0; FOR ( int i =0; is CONTLEN; i++) { Votos [:] = m Votos Br Categoria [act-][instituciós -1][i]; orderar (Mergrajes, votos): Mestoy asemiendo que va a orderar los nombres es funciós de cómo ordere los votos. cout « "Nombre del leguage mas elegido: « vlenguajes [0] « end/, Cout << "El porcertaje de los votos que: "« end]; For (int 1=0; 1 < CANTLEN; 1+4) \$ Cout << Votos [i] \* 100 / sumatoria << end; For (int 1=0; ICCANT\_ACT; i++) { 60 (int j=0; j < (AVT\_ NS; j++) { for (int K=0; K< CANT\_LEN; K++) { total + = mlores Por Categoria [;)[j][x]; if ( i == 0) { 3 alumnos += mVotos Por Categoria [:][i][k]i Cout ce contided total de encuestades: "<< total << eid! de alumnos: " << alumnos \* 100/tota << end); Cout << "Porcertage