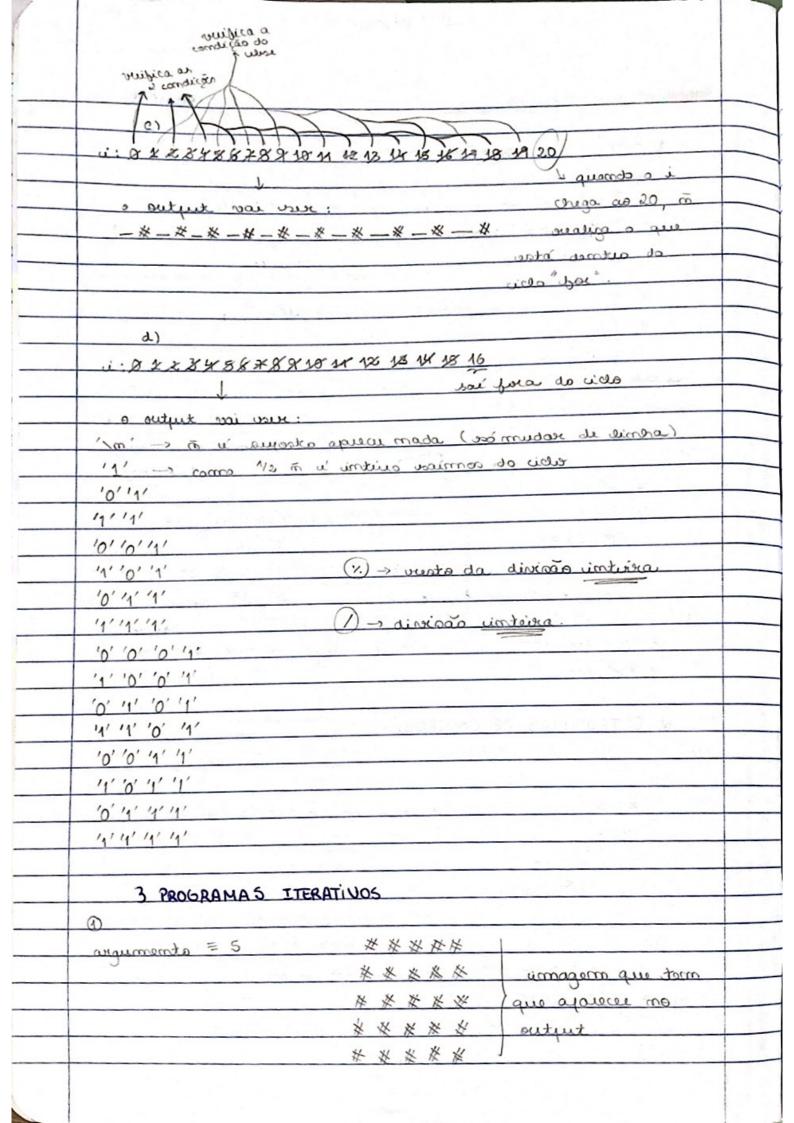
MOGRAMAÇÃO IMPERATIVA	FICHA 1
1 ESTADO E ATRIBUIÇÕES	
1 ESTADO E AMITO	-
0 2 12 ->	o output vai vaix: 12 16
y: 4 16	Kee & Cal War was a first to the
y:x	
(i)	o outfut vai invi: 0 ?
y: ?	o confine dos osoci o i)
m= quagnux c	
NoIA: amos ost contra	sumes o probrama no onk gago topo
notina, vai dax une	
0	Contract of the second of the
3 a: 65 66	
D: 32	
c: 4% 50 98	o output pai user.
C: 96 40 10	'A' 65 'B' 66 '2' 30
	'b' 98
(1)	
x: 200 300 100	> 8 output vai sove: 400 200
y:100 200	100 200
3.195	- M M. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1.
& ESTRUTURAS D	E CONTROLO
0	
as	The state of the s
x: 3	a output sai user:
y: 5	3 5
m se verifica	
a condição x>	THE CONTROL OF THE PARTY OF THE
c .	ajauntemente ceste cultimo
(ds)	1 jano elm conta.
x:8 x x 3 x 8 8	8789 101 m o output noi user:
4: \$ 2 3 6 10 15 2 winjing JJJJJ	128 36 48 55 66 11 66
wifica I) I)	
2.=11	duxou de
	vuibicae a



wild desemble Imagera (work a); wild desemble Imagera (e); ward ("xd"; &c); where a desemble Imagera (inta)? with a c; where a c; where a c; where a c; where a desemble a desemble a calcuma ("x"). I see sentice (2" and the color and color	* imagen (int m).
wint c; beauty (",d"; kc); directly (",d"; kc); directly (",d"; kc); putured directly transform (o); with e, c; for (e=0; e <m, (e="0," (ea="" e++);="" e<m,="" end="" exampl<="" example="" man="" of="" put="" th=""><th></th></m,>	
scamp ("xd"; kc); chreents tranger (e); putura d; mode disembatrager (inta)? mode disembatrager (inta)? mode disembatrager (inta)? par (l=0; l <m, ('#').="" ('m').="" (l="0;" (su="" (su<="" and="" apaix="" columa!!="" l++)?="" l<m,="" le="" par="" putabax="" scale="" td=""><td>void della () }</td></m,>	void della () }
scamp ("xd"; kc); chreents tranger (e); putura d; mode disembatrager (inta)? mode disembatrager (inta)? mode disembatrager (inta)? par (l=0; l <m, ('#').="" ('m').="" (l="0;" (su="" (su<="" and="" apaix="" columa!!="" l++)?="" l<m,="" le="" par="" putabax="" scale="" td=""><td>wint materi</td></m,>	wint materi
desembrationagema (c); quitarax putura 0; mod desembrationagema (insta)? mod desembrationagema (insta)? for (l=0; l <m, ('x').="" ('xm').="" (l="0;")="" \(="" a="" for="" l++)?="" l<m,="" of="" on="" on<="" que="" question="" quitance="" selection="" stem="" td=""><td></td></m,>	
meturan 8; noid desembatamagemo (innta)? unt l, c; for (l=0; l <m, ('m').="" (c="0," (su="" c++)?="" c<m,="" columa!!="" defeis="" e="" inline="" l++)?="" ma="" noice-nousea).<="" que="" question="" td="" tem="" you=""><td></td></m,>	
noid desembationagem (iento)? por (l=0; l <m, !!="" ('m').="" ('x').="" (c="0," (l="0;" (ou="" 1º="" c++)?="" c<m,="" columa="" columa<="" entra="" l++)?="" l<m,="" no="" por="" putchax="" son="" td=""><td>desemble smagerin (c); futchase</td></m,>	desemble smagerin (c); futchase
nord desembationagema (into)? int l.c; for (l=0; l <m, ('%').="" (l="0;" (su="" 1="" 2="" 3="" 4="" \forag<="" \forage="" \foragemate="" a="" and="" columna!!="" deserve="" l++)?="" l<m,="" noice-nousea).="" par="" putchax="" td="" yuamdo=""><td>vitura 0;</td></m,>	vitura 0;
and desembational gama (inta)? intal, c; for (l=0; l <m, (c="0," c++)?="" c<m,="" column="" l++)?="" of="" pr<="" properties="" que="" td="" the="" to=""><td></td></m,>	
int el, c; for (e=0; elm, e++); par (c=0, cm, c++); questão tem que columa! (su columa!! (su demagem que dem que ajaical mo que que que que que que que qu	void desembationagem (into)}
yer (c=0, c <m, (su="" c++)}="" columa!!="" de="" defeis="" ma="" que="" questão="" rece-versa).<="" td="" tem=""><td>umt e, c;</td></m,>	umt e, c;
yer (c=0, c <m, (su="" c++)}="" columa!!="" de="" defeis="" ma="" que="" questão="" rece-versa).<="" td="" tem=""><td>for (l=0; l<m, l++)}<="" td=""></m,></td></m,>	for (l=0; l <m, l++)}<="" td=""></m,>
yutchax ('%'). } yutchax ('%'). } yutchax ('\m'). \ yutchax ('\m')	() () () () () () () () () ()
Jumba e defeis ma columa!! (su vice-vouse).	Jutch are ('v')
esluma!! (su nice-nouse). 1 2 3 4 exquiments = 5	Line (1) -11
D 1 2 3 4 orguments = 5	a deles ma
arguments = 5	columa!! (su
arguments = 5	nice-nousa).
arguments = 5	
J → quando vé ½ - ½ - ½ que ajaiecer mo yar - ½ - ½ - ½ yar ajaiecer mo windar ½ imitende & istalio . is > wind disemberra (int m); uint main() } uint c; scamp ("Xd"; &c); desembrecia (c);	
J → quando vé ½ - ½ - ½ que ajaiecer mo yar - ½ - ½ - ½ yar ajaiecer mo windar ½ imitende & istalio . is > wind disemberra (int m); uint main() } uint c; scamp ("Xd"; &c); desembrecia (c);	exquenents = 5 & - X]
(y) → quando \(\lambda =	
yor - * - * - we output - youando .e	
J. quando & X - X - X umque * imelude & vatidio . in > word desembaccia (int m); int main() } int c; scamp ("Xd"; &c); desembrecia (c);	
impax * imclude & white . in > wind desemblecca (int m); int main() } int c; scamp ("Xd"; &c); desemblecca (c);) = anamaa 18
* imalia & istalio . is wild disembrecia (int m); int main() } int c; int c; scamp ("Xd"; &c); desembrecia (c);	
word desemblecea (int m); int main () } int c; scamp ("Xd"; &c); desemblecea (c);	amyaq
word desemblecea (int m); int main () } int c; scamp ("Xd"; &c); desemblecea (c);	V
uint c; uint c; uscamp ("Xd"; &c); desembrecia (c);	
uint c; scamp ("Xd"; &c); desembrecia (c);	word desembaccia (int m);
desembricia (c);	int main () }
desembricia (c);	uint c;
desembreca (c);	
,	
y ,	,
	i variation o;

```
desemprecia ( int m) }
int x=0; unt c, l;
       for (1=0; e < m; e++)}
          for (c=x, c< m+x,c++)}
            use } jutchase ("_");
          futchax ("1m");
argumento = 5
                      haizental:
 verkical:
* * * *
                   ③ ** * * * * * *
               (9) * * * * * * * * * * * * * * * *
* * * *
* * *
  void nutechar (intem, char c);
  roid restical (int m)}
     for ( i= 1; i < m; i++)}
       mutchare (ii, 'x');
        jutchar ('m');
     for (i=m; i>0; i--)}
       mutchar (ii, '*');
        jutionare ('Im');
```

```
void horizontal (int m)}
       foc ( l = 1; l <= m; l++)}
       mutchar (ii, 'k');
         jutchar ('\m'):
         u = u + 2;
  word myutchare (int m, chare c)}
     umt u;
for (i=0; i < m; i++)}
        jutchase (c);
1 (Revolução da aula!)
 argumento = eroio = 4
(int x)}
      int x, y;
 for (y=-x; y <= x; y++)}
      for (x = - se; x <= se; x ++)}
         if (x*x + y*y 1= vx x)}
         futchar ('*'); jutchar ('*').
         else } futchar (' '); futchar (' '); }
      jutionar ('Im');
```