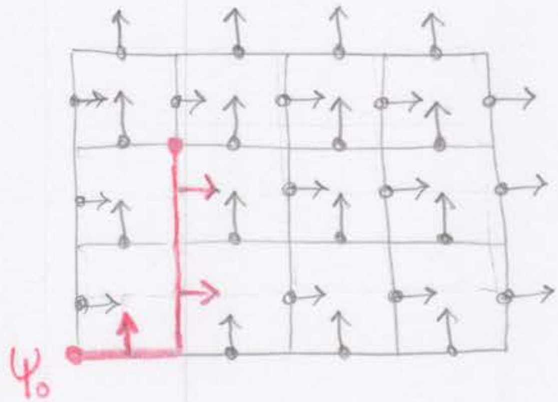
Computing streamlines from streamfunction

Definition: $\psi(x,y) = \psi_0(x_0,y_0) - \int_{x_0}^{x} v_y(x',y_0)\,dx' + \int_{y_0}^{y} v_x(x_0,y')\,dy'$



Given the location of the velocities on the faces, where is the natural location to evaluate ψ, given the definition of ψ above?

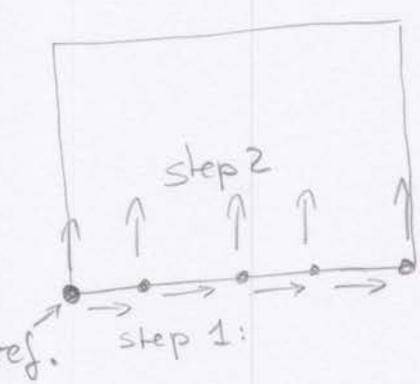
To compute ψ we need to integrate v_x & v_y along cell boundaries ⇒ evaluate ψ at <u>cell corners</u>!

Note, there is no need for numerical integration, because v_x and v_y are constant along each face.

<u>Hints on implementation:</u>

- The simple Riemann sum required to evaluate the integral is best implemented as a cummulative sum, available as cumsum.m in Matlab.
- Cumsum works also on matrices and you can specify the direction, ie. rows or columns

⇒ First you integrate along one boundary with a 1D cum sum
Then you integrate from the bnd into the domain along all faces at the same time by applying cumsum to a matrix of appropriately reshaped velocities.



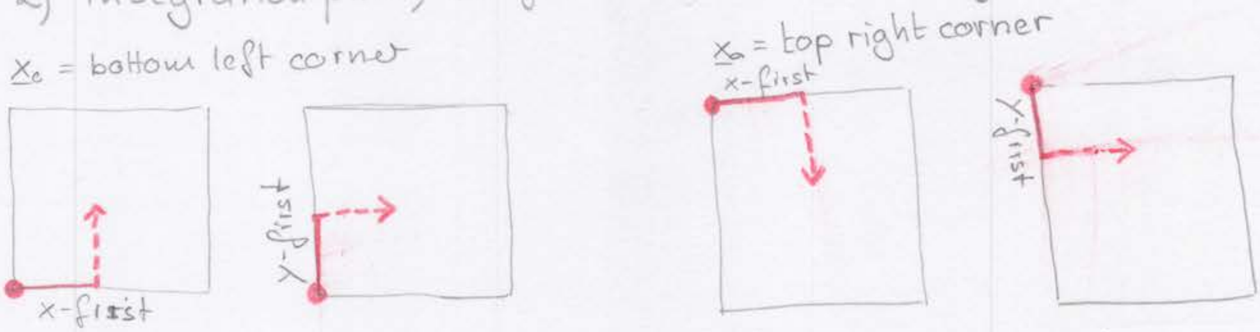
- Implemetation does <u>not</u> require solution of linear system ⇒ fast.

Note on path independence:

We have shown that ψ is uniquely defined, up to a constant ψ_0.

In our numerical calculation we have to choose

1) starting point $\underline{x}_0$
2) integration path, ie. first in x-dir then in y-dir or vice versa.



The choice of the starting point, $\underline{x}_0$, affects the constant, ψ_0, if there is flow accross the boundary.

The choice of the integration path does not $\Rightarrow$ path independence.

Once ψ is known, the streamlines are easily plotted as contours of the stream function. Use evenly spaced contours so that the spacing of the streamlines represents the velocity of the flow.

Finally, if you have an interesting flow field plot the actual stream function, it is often instructive.