Title of the document

Your name(s) here
(Dated: September 4, 2021)

List a link to your github repository here!

PROBLEM 1

https://github.com/henrikbreitenstein/FYS3150.git

PROBLEM 1

Poisson equation

$$-\frac{\mathrm{d}^2u}{\mathrm{d}x^2} = f(x)$$

Replaceing $f(x)$ with given function

$$\begin{aligned} -\frac{\mathrm{d}^2u}{\mathrm{d}x^2} =& 100e^{-10x} \\ -\mathrm{d}^2u =& 100e^{-10x}\ \mathrm{d}x^2 \end{aligned}$$

Taking integrals

$$\begin{aligned} -\int\int \mathrm{d}^2u &= \int\int 100e^{-10x}\ \mathrm{d}x^2 \\ -u &= \int -10e^{-10x} + c_1\ \mathrm{d}x \\ -u =& e^{-10x} + c_1x + c_2 \\ u =& -e^{-10x} - c_1x - c_2 \end{aligned}$$

Using initial conditions:

$$u(0) = 0 \Rightarrow -1 - c_2 = 0 \tag{1}$$

$$u(1) = 0 \Rightarrow -e^{-10} - c1 - c2 = 0 \tag{2}$$

With 1 and 2 we get:

$$\begin{aligned} c2 =& -1 \\ c1 =& 1 - e^{-10} \end{aligned}$$

By replacing c_1 and c_2 we get:

$$u = 1 - (1 - e^{-10})x - e^{-10x} \tag{3}$$

PROBLEM 2

We write equations using the LaTeX `equation` (or `align`) environments. Here is an equation with numbering

$$\mathbf{F} = \frac{\mathrm{d}\mathbf{p}}{\mathrm{d}t}, \tag{4}$$

and here is one without numbering:

PROBLEM 5

A. Problem b

Av $A\vec{v} = \vec{g}$ vil vi finne alle verdier mellom grensebetingelsene $u(0) = u(1) = 0$. $\vec{v*}$ er den samme som $\vec{v}$ men vi har lagt til v_0 og v_{n+1} som er grensebetingelsene.

I. PROBLEM 6

A

Vi har nå en vektor

$$\vec{v} = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix}$$

og en $n \times n$-matrise

$$\mathbf{A} = \begin{Bmatrix} b_1 & c_2 & 0 & 0 & \cdots & 0 & 0 \\ a_2 & b_2 & c_2 & 0 & \cdots & 0 & 0 \\ 0 & a_3 & b_3 & c_3 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & \cdots & \cdots & \cdots & a_n & b_n \end{Bmatrix}$$

Matrisemultipliserer vi disse får vi

$$\mathbf{A}\vec{v} = \begin{array}{cccccc} (I) & b_1v_1 & +c_1v_2 & & & = g_1 \\ (II) & a_2v_1 & +b_2v_2 & +c_3v_3 & & = g_2 \\ (III) & & a_3v_2 & +b_3v_3 & +c_3v_4 & = g_3 \\ \vdots & & & & & \\ (n) & & & a_nv_{n-1} & +b_nv_n & = g_n \end{array}$$

Herfra skal vi radredusere og starter med å ta $(II) - \frac{a_2}{b_1}(I)$ slik at vi får

$$(II*) \;\; 0 \;\; b_2 - \tfrac{c_1 \cdot a_2}{b_1} \;\; c_2 \;\; \cdots \;\; g_2 - g_1 \tfrac{a_2}{b_1}$$

og vi ser da at a_2 går bort. Vi kan også sette $b_2^* \equiv b_2 - \frac{a_2 \cdot c_1}{b_1}$ og $g_2^* \equiv g_2 - g_1 \frac{a_2}{b_1}$. Da har vi mellom (II^*) og (III) noe som ser ganske likt ut som det vi hadde mellom (I) og (II). Derfor gjør vi det samme som vi gjorde før og tar $(III) - \frac{a_3}{b_2^*}(II)^*$ og får

$$(III)^* \;\; 0 \;\; 0 \;\; b_3 - \tfrac{c_2 \cdot a_3}{b_2^*} \;\; c_3 \;\; g_3 - g_1 \tfrac{a_3}{b_2^*}$$

og vi kan igjen definere $b_3^* = b_3 - \frac{c_2 \cdot a_3}{b_2^*}$ og $g_3^* = g_3 - g_1 \frac{a_3}{b_2^*}$. Og vi kan da fortsette med dette nedover som $(k) - \frac{a_k}{b_{k-1}^*}(k-1)^*$. Så har vi fjernet a-ene så da må vi fjerne c-ene. Vi starter nå på siste og neste siste rad, altså $(n)^*$ og $(n-1)^*$ som nå er

$$\begin{array}{cccccc} (n-1)^* & 0 & \cdots & b_{n-1}^* & c_{n-1} & g_{n-1}^* \\ (n)^* & 0 & \cdots & 0 & b_n^* & g_{n-1}^* \end{array}$$

så hvis vi da tar $(n-1)^* - \frac{c_{n-1}}{b_n^*}(n)^*$ får vi

$$(n\tilde{-}1) \;\; 0 \;\; \cdots \;\; b_{n-1}^* \;\; 0 \;\; g_{n-1}^* - \frac{c_{n-1}g_n^*}{b_n^*}$$

og vi setter $g_{n\tilde{-}1} = g_{n-1}^* - \frac{c_{n-1}g_n^*}{b_n^*}$ og dette gjør vi videre oppover som $(k-1) - \frac{c_k}{b_{k-1}}(k)$ Til slutt står vi da bare igjen med b^*-ene og disse kan vi da dele på seg selv og vi får en identitetsmatrise. Vi kan nå skrive dette som en algoritme. Anta vi har en tridiagonal matrise

$$A = U = \begin{pmatrix} u_{1,1} & u_{1,2} & \cdots & u_{1,n} \\ u_{2,1} & u_{2,2} & \cdots & u_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ u_{n,1} & un,2 & \cdots & un,n \end{pmatrix}$$

og en som skal løses for vektoren

$$g = h = \begin{pmatrix} h_1 \\ h_2 \\ \vdots \\ h_n \end{pmatrix}$$

Algorithm 1 Radredusering av tridiagonal matrise

for $i = 2, 3, ..., n$ **do** ▷ Forward substitution, $n - 1$ repetisjoner

$$t \leftarrow \frac{u_{i,i-1}}{u_i - 1, i}$$

▷ 1 FLOP

$$u_{i,i-1} \leftarrow u_{i,i-1} - u_{i-1,i-1} \cdot t$$

▷ 2 FLOPs

$$u_{i,i} \leftarrow u_{i,i} - u_{i-1,i-1} \cdot t$$

▷ 2 FLOPs

$$h_i \leftarrow h_i - h_{i-1} t$$

▷ 2 FLOPs

▷ Til sammen $7 \cdot (n-1)$ FLOPs i loopen

for $j = n - 1, n - 2, ..., 1$ **do** ▷ Backward Substitution, $n - 1$ repetisjoner

$$k \leftarrow \frac{u_{j,j+1}}{u_{j+1,j}}$$

▷ 2 FLOPs

$$u_{j,j+1} \leftarrow u_{j,j+1} - u_{j+1,j+1} \cdot k$$

▷ 2 FLOPs

$$u_{j,j} \leftarrow u_{j,j} - u_{j+1,j} \cdot k$$

▷ 2 FLOPs

$$h_j \leftarrow h_j - h_{j+1} \cdot k$$

▷ 2 FLOPs

▷ Til sammen $7 \cdot (n-1)$ FLOPs i loopen

for $l = 1, 2, \cdots n$ **do** ▷ Deler for å få identitetsmatrise, til sammen n repetisjoner

$$u_l \leftarrow \frac{u_l}{u_l}$$

▷ 1 FLOP

$$h_l \leftarrow \frac{h_l}{u_l}$$

▷ 1 FLOPs

▷ Til sammen $2 \cdot n$ FLOPs

Da ser vi at vi til sammen får $2 \cdot n + 2 \cdot 7 \cdot (n-1) = 16n - 14$ FLOPs.

Sometimes it is useful to refer back to a previous equation, like we're demonstrating here for equation 4.

We can include figures using the `figure` environment. Whenever we include a figure or table, we *must* make sure to actually refer to it in the main text, e.g. something like this: "In figure **??** we show ...".

Also, note the LaTeX code we used to get correct quotation marks in the previous sentence. (Simply using the `"` key on your keyboard will give the wrong result.) Figures should preferably be vector graphics (e.g. a `.pdf` file) rather than raster graphics (e.g. a `.png` file).

By the way, don't worry too much about where LaTeX decides to place your figures and tables — LaTeX knows more than we do about proper document layout. As long as you label all your figures and tables and refer to them in the text, it's all good. Of course, in some cases it can be worth trying to force a specific placement, to avoid the figure/table appearing many pages away from the main text discussing it, but this isn't something you should spend time on until the very end of the writing process.

Next up is a table, created using the `table` and `tabular` environments. We refer to it by table I.

Finally, we can list algorithms by using the `algorithm` environment, as demonstrated here for algorithm 1.

Number of points	Output
10	0.3086
100	0.2550

TABLE I. Write a descriptive caption here, explaining the content of your table.