

Delrapport 4

Bassänger/Solfångare

Axel Gustafsson, Rasmus Tiberg

Bassänger/Solfångare	1
1. Bakgrund	2
1.1. Allmänt.....	2
1.2. Solfångare	2
1.2.1. Plana Solfångare	2
1.2.2. Vakuum Solfångare.....	2
1.2.3. Jämförelser	2
1.3. Geotermisk lagring.....	3
1.4 Bassänger.....	3
1.5. Överdragningsställe.....	3
2. Material och metoder.....	3
2.1 Modellering av poler.	3
3. Resultat 50m 7-10 kubik /dygn. Äventyr 15-20 kubik,	6
3.1. Bassängmodeller	6
Referenser.....	8
Appendix.....	10

1. Bakgrund

1.1. Allmänt

Efter ett besök på anläggningen studerades bakomliggande system för uppvärmning av vatten till bassängerna. Det framgick då att uppvärmningen av bassängerna endast drevs av värme från Uppsalas Fjärrvärmenät. Eftersom uppvärmning av bassängerna är en väldigt energikrävande process upplever Uppsala Arenor och fastigheter att stora delar av deras energitillförsel kommer från Vattenfalls fjärrvärmenät i uppsala. Med tanke på detta kan det vara rimligt att studera flera möjligheter för uppvärmning eller effektivisering av värmeflödena för bassängerna.

1.2. Solfångare

Idag finns det en hel del nytänkande ideer och uppfinningar som går att implementera inom badhusmiljöer. En sådan teknik är solfångare som använder sig av solenergi för att värma upp bassängerna. Grundprincipen bygger på att värma upp vattenflöden genom en viss solfångare som utnyttjar solens strålar. Inom denna teknik används två olika typer av solfångare dessa är *Plana-* Och *Vakuumsolfångare*.

1.2.1. Plana Solfångare

De plana solfångarna är en enklare teknik som använder sig av en absorбатыryta där solens strålar värmer vatten eller en vattenblandning med glykol. Denna absorбатыryta är ofta svartfärgad för att få en högre absorption. I absorбатыrytan cirkulerar vattenflöden som senare kan transporteras till en värmelagring eller direkt in i ett system som utnyttjar vattnets värme. En av fördelarna med plana solfångare är att de bygger på enklare teknik till ett lite billigare pris. (Ayompe et al., 2011)

1.2.2. Vakuum Solfångare

Vakuumsolfångare bygger på parallella vakuumglastuber där varje parallell tub har en egen absorбатыryta. Värmen byggs upp i tuberna och i den varmaste delen överförs värmen via värmeväxlare som kan kopplas vidare till ett varmvattensystem. (Khalid Almutairi et al., 2021) Som Exempel kan denna värme kopplas vidare till ett system som bidrar med värme till simbassänger.

1.2.3. Jämförelser

I en studie gjorde man generella jämförelser i faktatabeller. Där nämner man att absorptionen för Vakuum-solfångare kan sträcka sig upp mot 95% medan de plana solfångarna som högst kan nå 80% absorption. (Khalid Almutairi et al., 2021) Var det gäller värmeförluster verkar vakuum-solfångarna ha mindre förluster på 6% i jämförelse med de planas förluster på mer än 20%. Ytterligare påpekar man att underhållet för en plan solfångare ofta är mer komplicerat och kräver professionell personal. Samtidigt kan fall av läckage eller skada kräva att fångaren behöver bytas ut i sin helhet. Vakuum-solfångarna har däremot ett enklare underhåll och har en bättre möjlighet för reparation eller byte av komponenter. Några ytterligare fördelar med vakuum-solfångare är den kan överföra värme upp till 7°C/h samt att den har låg risk för frysskador. (Khalid Almutairi et al., 2021)

1.3. Geotermisk lagring

Eftersom solfångare är direkt kopplade till solens tillgångar och det svenska klimatet kommer det finnas stora skillnader i energiutvinningen beroende på årstiden. Med tanke på detta kan det vara rimligt att diskutera lagringsalternativ för varmvatten. Om det skulle vara möjligt kan man jämnna ut värmebehoven med de resurser som finns att tillgå. En teknik som redan är väl känd och använd är geotermisk lagring.

Det finns borrhålslager som utnyttjar berggrundens värmeledningsförmåga. Enligt (Erlström, et al., 2016) kan man med ett flertal borrhålslager skapa man möjligheten att säsongsvara uppvärmt vatten. Ett sådant lager kan utnyttjas tillsammans med solfångare för att skapa en mer pålitlig värmeförsörjning året runt. Med tanke på Simhallens värmeberoende under alla årstider skulle detta kunna vara högst aktuellt

1.4 Bassänger

Stora vattensamlingar avger mycket värme. De avger genom konduktion konvektion och radiation. Vatten avdunstar även vid rumstemperatur och både vattnet och lufttemperaturen inom simhallar är över det. Ca 26-30 °C där lufttemperaturen är 3 °C däröver. Detta leder till att konvektionen i inomhusimhallar ökar värmen. I polen. För att kompensera för den avgivna värmen använder man inom simhallar värmeväxlare för att öka temperaturen på det inflydande vattnet som inom Fyrishovs anläggningar är mellan 10-15 °C. De bassänger som studeras är 50-metersbassängen på 28 °C, äventyrsbadet på 30 °C samt utomhuspolen på 26 °C. Att få en överblick över hur de olika bassängernas värmesystem fungerar kan sedan effektiva lösningar appliceras för att minska förlusttoppar.

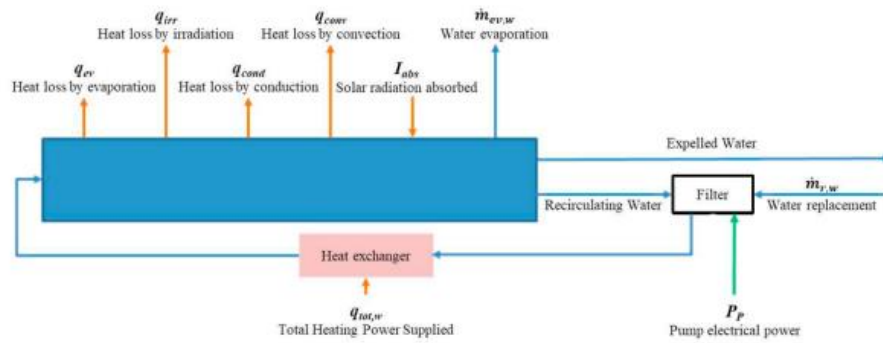
1.5. Överdragningsställe

Då inomhustemperaturen hålls konstant så är det inget behov att utforska möjligheter att åtgärda beroende på lufttemperatur. Detta är dock möjligt för utomhuspolen. Då lufttemperaturen sjunker under nätterna och det inte vistas besökare är en energisparande åtgärd att applicera täckning på nätterna. Detta minskar avdunstningen med 95%, konvektionen minskar därmed med 85% och radiationsförlusterna minskar med 70%. Att undersöka huruvida överdragningsställe över 14 h per sommardag kan påverka värmeförlusterna från utomhuspolen.

2. Material och metoder

2.1 Modellering av poler.

För att kartlägga energianvändningen för uppvärmning av polerna behövs de modelleras i ett program där relevant data kan införas för att ge ett resultat. Beräkning av polernas dimensioner ges av angivna data alternativt egna mätningar från skalenliga ritningar. Modellen kommer att följa systembeskrivningen från (Orecchini F et al 2017).



Figur 1: Massa och energibalans för en simbassäng

https://www.researchgate.net/publication/319995175_Energy_analysis_of_swimming_pools_for_sports_activities_cost_effective_solutions_for_efficiency_improvement

Där de olika förlusterna beskrivs genom följande formler.

Vattnets avdunstningsenergi alltså den energi som försvinner då vattnet i bassängen avdunstar ges av

- $q_{ev} = h_{ev} \cdot A \cdot (P_{sat} - P_{air})$

(1)

- där

P_{sat}

ges av Tetens tryckformel

$$610.78 e^{17.27 - T_c / T_c + 237.5}$$

(2)

Strålningsvärmeförlusten ges av Stefan Boltzmanns strålningsförlustlag. Detta agerar som värmetillförsel för inomhuspöler då lufttemperatur hålls högre än polens temperatur.

- $q_{irr} = \epsilon \cdot \sigma \cdot A \cdot (T_{4pool}^4 - T_{4air}^4)$

(3)

Värmeförlusten till polens botten är angivet genom konduktionslagen

- $q_{cond} = k_d \cdot A \cdot (T_{pool} - T_{ground})$

(4)

Konvektionen mellan luften och vattnet som är värmeförlust till polen då lufttemperatur hålls konstant högre än bassängtemperatur.

- $q_{\text{conv}} = h \cdot A \cdot (T_{\text{pool}} - T_{\text{air}})$

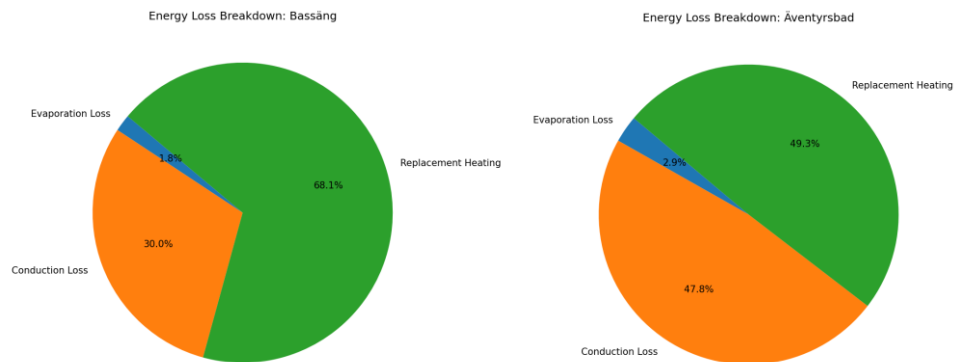
Värmen som krävs för att värma det inkommande ersättningsvattnet i bassängen.

- $q_{\text{rep}} = c \cdot \rho \cdot m \cdot (T_{\text{pool}} - 288.15)$

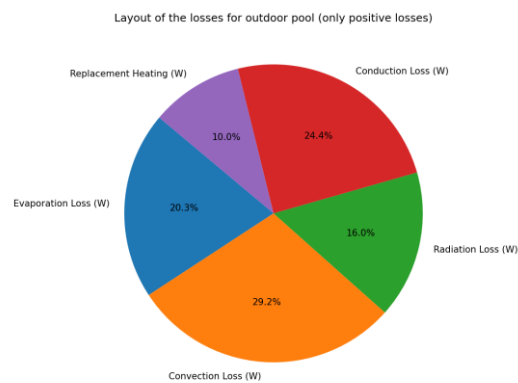
Dessutom så kommer en av polerna vara utomhuspoolen där lufttemperaturen varierar med data från SMHI över öppningstidsveckorna 2024. För beräkning av värmeförlust från solstrålningen används liknande data från SMHI dock är det solinstrålning från Borlänge då uppsala saknar solmätning. För denna del har flera antaganden skett för olika värden till exempel h och k värdena. Även vissa mått har antagits både djupet av vissa bassänger då djupet varierar samt djup på betongen under polen.

3. Resultat 50m 7-10 kubik /dygn. Äventyr 15-20 kubik,

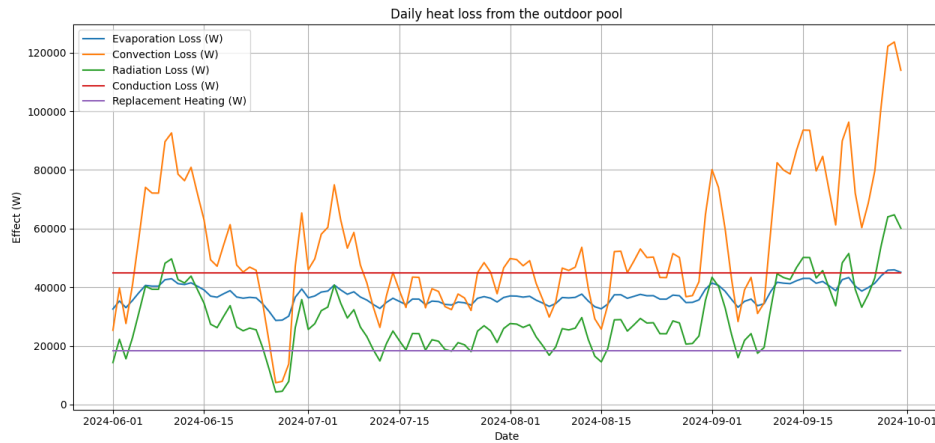
3.1. Bassängmodeller



Figur 2: Pie chart över de största värmeförlusterna för inomhuspoolerna



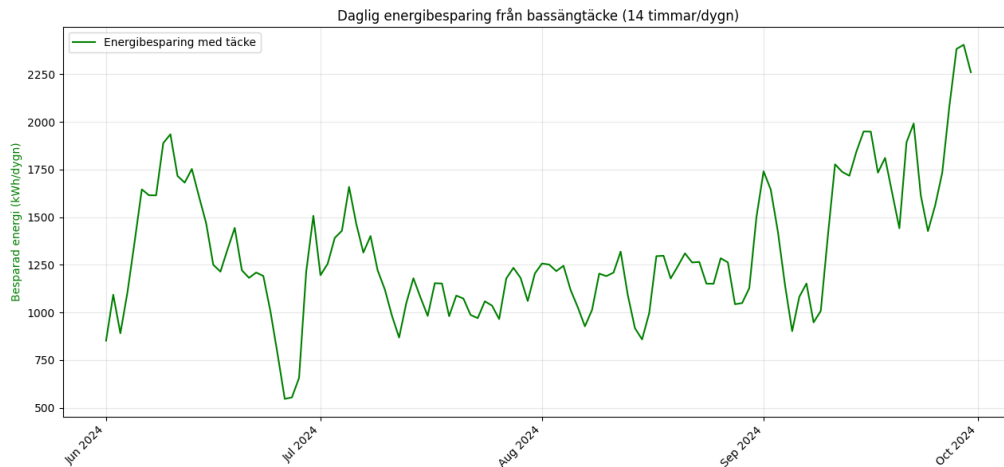
Figur 3: Pie chart över medelförlusterna för utomhuspoolen



Figur 4: Daglig energiförlust för utomhuspoolen

Figur 5: Solabsorbtion mot summan av värmeförlusterna

Genomsnittlig effektvinst från solfångare: 2782.44 W innebär att $\Delta T = \frac{Q}{\dot{m} \cdot c} \approx 2,5 \text{ K}$.
Solfångare bör kunna bidra till 2,5 graders uppvärmning.



Figur 6: Besparing av energi via ett övertäckningssystem.

Referenser

Teli, D., Bourikas, K. and James, P.A.B. (2017) 'Energy analysis of swimming pools for sports activities: cost-effective solutions for efficiency improvement', *Energy and Buildings*, 151, pp. 318-331. Available at: <https://www.researchgate.net/publication/319995175> (Accessed: [04-18-25]).

SMHI (2024) Ladda ner väderobservationer - *Lufttemperatur* . Available at: <https://www.smhi.se/data/hitta-data-for-en-plats/ladda-ner-vaderobservationer/airtemperatureInstant> (Accessed: 27 April 2025).

SMHI (2024) Solstrålning - *Irradians*. Available at: <https://www.smhi.se/data/solstralning/solstralning/irradiance/105285> (Accessed: 18 April 2025).

Northeast Pool Covers (2024) *Energy & Heating Benefits*. Available at: <https://northeastpoolcovers.com/cover-benefits/energy-heating-benefits/> (Accessed: 3 May 2025).

U.S. Department of Energy (no date) *Swimming Pool Covers*. Available at: <https://www.energy.gov/energysaver/swimming-pool-covers> (Accessed: 5 May 2025).

ASHRAE (2019) *ASHRAE Handbook - HVAC Applications Applications SI Swimming Pool: Swimming Pools*. Available at: <https://www.scribd.com/document/588317610/2019-ASHRAE-Handbook-HVAC-Applications-SI-swimming-pool> (Accessed: 5 May 2025).

Ayompe, L.M. et al. (2011) Comparative field performance study of flat plate and heat pipe evacuated tube collectors (ETCs) for domestic water heating systems in a temperate climate. Available at: <https://www-sciencedirect-com.ezproxy.its.uu.se/science/article/pii/S0360544211002003> (Accessed: 3 May 2025)

Erlström, M. et al. (2016) Geologisk information för geoenergianläggningar – en översikt. Available at: <https://www.sgu.se/globalassets/samhallsplanering/energi/geologisk-information-for-geoenergianlaggningar.pdf> (Accessed: 20 April 2025).

Khalid Almutairi et al. (2021) Techno-Economic Investigation of Using Solar Energy for Heating Swimming Pools in Buildings and Producing Hydrogen: A Case Study. Available at: <https://www.frontiersin.org/journals/energy-research/articles/10.3389/fenrg.2021.680103/full> (Accessed: 15 April 2025).

Uppsala arenor och fastigheter (2025a) Renovering av Fyrishov för en längre livslängd. Available at: <https://arenorochfastigheter.uppsala.se/projekt-utveckling-underhall/renovering-av-fyrishov-for-en-langre-livslangd/> (Accessed: 3 March 2025).

Uppsala arenor och fastigheter (2025b) Stort behov av nya badanläggningar i Uppsala. Available at: <https://arenorochfastigheter.uppsala.se/projekt-utveckling-underhall/nya-badanlaggningar/> (Accessed: 3 March 2025).

Uppsala Kommun (2025) Kommunen satsar på Fyrishov till 2045. Available at:
<https://www.uppsala.se/kommun-och-politik/nyheter-och-pessmeddelanden/2025/kommunen-satsar-pa-fyrishov-till-2045/> (Accessed: 3 March 2025).

Appendix

```
import math
import matplotlib.pyplot as plt
import pandas as pd

class SwimmingPoolEnergyAnalysis:
    def __init__(self, pool_length, pool_width, avg_depth, T_pool, T_air, T_ground,
RH_air,
        solar_radiation, replacement_rate, pump_head, pump_efficiency,
pump_flow_rate=None,
        indoor=True):
        self.pool_length = pool_length
        self.pool_width = pool_width
        self.avg_depth = avg_depth
        self.T_pool = T_pool
        self.T_air = T_air
        self.T_ground = T_ground
        self.RH_air = RH_air
        self.solar_radiation = solar_radiation
        self.replacement_rate = replacement_rate
        self.pump_head = pump_head
        self.pump_efficiency = pump_efficiency
        self.indoor = indoor
        self.pump_flow_rate = pump_flow_rate if pump_flow_rate else self.volume() / (6 *
3600)

        # Constants
        self.rho_water = 1000
        self.c_water = 4186
        self.g = 9.82
        self.sigma = 5.67e-8
        self.absorption_coeff = 0.85

        # Heat transfer coefficients
        self.h_evap = 6 if self.indoor else 25
        self.h_conv = 10
        self.k_ground = 1.5
        self.ground_thickness = 0.3

    def surface_area(self):
        return self.pool_length * self.pool_width

    def volume(self):
        return self.pool_length * self.pool_width * self.avg_depth
```

```

def pump_power(self):
    return (self.pump_head * self.pump_flow_rate * self.rho_water * self.g) /
self.pump_efficiency

def saturation_pressure(self, T):
    return 610.78 * math.exp(17.27 * (T - 273.15) / (T - 35.85))

def evaporation_loss(self):
    A = self.surface_area()
    P_sat = self.saturation_pressure(self.T_pool)
    P_air = self.RH_air * self.saturation_pressure(self.T_air)
    return self.h_evap * A * (P_sat - P_air) / 1000

def convection_loss(self):
    return self.h_conv * self.surface_area() * (self.T_pool - self.T_air)

def radiation_loss(self):
    emissivity = 0.95
    return emissivity * self.sigma * self.surface_area() * (self.T_pool**4 - self.T_air**4)

def conduction_loss(self):
    return (self.k_ground / self.ground_thickness) * self.surface_area() * (self.T_pool -
self.T_ground)

def solar_absorption(self):
    return self.solar_radiation * self.surface_area() * self.absorption_coeff

def replacement_heating(self):
    return self.c_water * self.replacement_rate * self.rho_water * (self.T_pool - 288.15)

def total_heating_power(self):
    q_ev = self.evaporation_loss()
    q_conv = self.convection_loss()
    q_rad = self.radiation_loss()
    q_cond = self.conduction_loss()
    I_abs = self.solar_absorption()
    q_replacement = self.replacement_heating()
    return q_ev + q_conv + q_rad + q_cond - I_abs + q_replacement

def energy_loss_breakdown(self):
    return {
        "Pump Power (W)": self.pump_power(),
        "Evaporation Loss (W)": self.evaporation_loss(),
        "Convection Loss (W)": self.convection_loss(),

```

```

        "Radiation Loss (W)": self.radiation_loss(),
        "Conduction Loss (W)": self.conduction_loss(),
        "Solar Absorption (W)": self.solar_absorption(),
        "Replacement Heating (W)": self.replacement_heating(),
        "Total Heating Power (W)": self.total_heating_power()
    }

def plot_energy_pie(name, breakdown, indoor=True):
    labels, values = [], []
    for key, value in breakdown.items():
        # Skip pump power, total heating power, solar absorption (gain), and negative values
        (gains
         if ("Pump Power" in key or
            "Total Heating Power" in key or
            "Solar Absorption" in key or
            value <= 0):
            continue
        labels.append(key.replace(" (W)", ""))
        values.append(abs(value))
    plt.figure(figsize=(8, 6))
    plt.pie(values, labels=labels, autopct='%1.1f%%', startangle=140)
    plt.title(f"Energy Loss Breakdown: {name}")
    plt.tight_layout()
    plt.show()

def las_in_utomhustemp(filnamn):
    df = pd.read_csv(filnamn, sep=';', skiprows=6, encoding='utf-8',
                     names=["Datum", "Tid", "Lufttemperatur", "Kvalitet", "Tom1", "Tom2"])
    df = df[["Datum", "Tid", "Lufttemperatur"]]
    df["Timestamp"] = pd.to_datetime(df["Datum"] + " " + df["Tid"], errors='coerce')

    df["Lufttemperatur"] = pd.to_numeric(df["Lufttemperatur"], errors='coerce')
    df = df.dropna().set_index("Timestamp")
    df_dag = df["Lufttemperatur"].resample("D").mean()
    return df_dag + 274.15

def simulate_outdoor_pool_over_time(temperatures_K, pool_config):
    results = []
    for i, T_air in enumerate(temperatures_K):
        pool = SwimmingPoolEnergyAnalysis(**{**pool_config, "T_air": T_air})
        heating_power = pool.total_heating_power()
        results.append((i, T_air - 273.15, heating_power))
    return results

if __name__ == "__main__":

```

```

# Inomhuspooler
main_pool = SwimmingPoolEnergyAnalysis(
    pool_length=50, pool_width=25, avg_depth=2.88,
    T_pool=301.15, T_air=303.15, T_ground=285.15,
    RH_air=0.7, solar_radiation=0, replacement_rate=0.00417,
    pump_head=12, pump_efficiency=0.8, indoor=True
)
adventure_pool = SwimmingPoolEnergyAnalysis(
    pool_length=25, pool_width=15.6, avg_depth=1.28,
    T_pool=303.15, T_air=305.15, T_ground=285.15,
    RH_air=0.7, solar_radiation=0, replacement_rate=0.000577,
    pump_head=12, pump_efficiency=0.8, indoor=True
)

for name, pool in [("Bassäng", main_pool), ("Äventyrsbad", adventure_pool)]:
    print(f"\n--- {name} ---")
    results = pool.energy_loss_breakdown()
    for key, value in results.items():
        print(f"{key}: {value:.2f}")
    plot_energy_pie(name, results, indoor=True)

tempfil = r"C:\Users\axelg\Downloads\smhi-opendata_1_97510_202406_202409.csv"
outdoor_temperatures_K = las_in_utomhustemp(tempfil)
outdoor_pool_config = {
    "pool_length": 20,
    "pool_width": 32,
    "avg_depth": 1.4,
    "T_pool": 299.15,
    "T_ground": 285.15,
    "RH_air": 0.5,
    "solar_radiation": 250,
    "replacement_rate": 0.0004,
    "pump_head": 9,
    "pump_efficiency": 0.77,
    "indoor": False
}

results = simulate_outdoor_pool_over_time(outdoor_temperatures_K,
outdoor_pool_config)
print("\n=== Utomhuspool: Simulerat värmebehov ===")
for timestep, T_air_C, power in results:
    print(f"Timestep {timestep:2d} | T_air = {T_air_C:.1f} °C | Heating Power =
{power:.2f} W")

```

```

loss_types = ["Evaporation Loss (W)", "Convection Loss (W)", "Radiation Loss (W)",
              "Conduction Loss (W)", "Replacement Heating (W)"]
loss_time_series = {key: [] for key in loss_types}
timestamps = []

for i, T_air in enumerate(outdoor_temperatures_K):
    pool = SwimmingPoolEnergyAnalysis(**{**outdoor_pool_config, "T_air": T_air})
    losses = pool.energy_loss_breakdown()
    for key in loss_types:
        loss_time_series[key].append(losses[key])
    timestamps.append(outdoor_temperatures_K.index[i])

plt.figure(figsize=(12, 6))
for key in loss_types:
    plt.plot(timestamps, loss_time_series[key], label=key)
plt.title("Daily heat loss from the outdoor pool")
plt.xlabel("Date")
plt.ylabel("Effect (W)")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()

# Beräkna genomsnittliga förluster (exklusive negativa värden och solar absorption)
average_losses = {key: sum(values)/len(values) for key, values in
loss_time_series.items()}
average_losses = {k: v for k, v in average_losses.items() if v > 0} # Remove negative
values

plt.figure(figsize=(8, 8))
plt.pie(average_losses.values(), labels=average_losses.keys(), autopct="%1.1f%%",
startangle=140)
plt.title("Layout of the losses for outdoor pool (only positive losses)")
plt.tight_layout()
plt.show()

```